IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

T. S. Huang

K. S. Fu

Contractor: Purdue University
Contract No. F30602-75-C-0150
Effective Date of Contract: 1 November 1975
Contract Expiration Date: 31 October 1976
Amount of Contract: $350,000.00
Program Code Number: 5D30
Period of work covered: February - April 1976

Principal Investigator: T. S. Huang
K. S. Fu

Phone: 317 493-3361

Project Engineer: David J. Brazil, Capt, USAF
Phone: 315 330-3175

Approved for public release;
distribution unlimited.

This research was supported by the Defense Advanced Research Projects Agency of the Department of Defense and was monitored by Capt. David J. Brazil (IRRO), Griffiss AFB NY 13441

TABLE OF CONTENTS

IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

ABSTRACT

This report summarizes the results of our research program on Image Understanding and Information Extraction supported by the Defense Advanced Research Projects Agency under Contract F30602-75-C-0150. The report covers the period February 1 to April 30, 1976.

The objective of our research is to achieve a better understanding of image structure and to use this knowledge to develop techniques for image analysis and processing tasks, especially information extraction. Our emphasis is on syntactic decomposition and recognition of imagery based on scene analysis. It is our hope that the results of this research will form the basis for the development of technology relevant to military applications of machine extraction of information from aircraft and satellite imagery.

IMAGE UNDERSTANDING AND INFORMATION EXTRACTION

Research Summary

This report summarizes our research progress during the period February 1 to April 30, 1976, in Image Understanding and Information Extraction.

Our research objective is to achieve a better understanding of image structure and to use this knowledge to develop techniques for information extraction from imagery. It is our hope that the results of this research will form the basis for the development of technology relevant to military applications of machine extraction of information from aircraft and satellite imagery.

Our research projects fall into five heavily overlapping areas: Image Segmentation, Image Attributes, Image Structure, Image Recognition Techniques, Preprocessing, and Applications.

IMAGE SEGMENTATION - We pursue two approaches to image segmentation: edge detection, and region growing. In edge detection, the thrust of our research is to use syntactic methods to help edge detection in noisy and blurred imagery. As a step along that direction, we are studying characteristics of digital edges. Almost all past works on digital edges and curves dealt with the ideal situation, while we are interested mainly in real-life digitized images. Our experimental results, reported by Tang and Huang, indicate that the properties of real-life digital straight edges are quite different from those of ideal edges. In the same report, a technique of recognizing real-line digital straight edges is also presented.

A by-product of our edge detection research is a technique of accurately estimating edge locations which holds great promise in mensuration applications. This technique, reported by Burnett and Huang, is based on the Viterbi algorithm. It can take account of arbitrary edge profiles and film noise and is computationally simple.

In region growing, we are studying both region merging techniques and different similarity measures among regions.

IMAGE ATTRIBUTES - We are doing both texture and shape analysis. A new class of texture descriptors, the max-min descriptors, has been developed. These descriptors are very simple to compute and perform extremely well in various classification problems. We are currently exploiting the use of max-min descriptors in texture boundary detection and region growing.

In shape analysis, we have done extensive study on Fourier boundary descriptors and are striving to improve their performance. We are also developing grammars for various classes of objects of military significance, such as airports, and tanks.

IMAGE STRUCTURE - The thrust of our research in this area is to use syntactic methods to do scene analysis. Fu and Li report on the use of tree grammar in detecting highways and rivers in LANDSAT imagery. Tree grammar was also used in helping scene segmentation (Fu and Keng).

IMAGE RECOGNITION TECHNIQUES - We pursue two topics: the use of branch and bound techniques in solving recognition problems, and the use of context in statistical classification. In Swain and Yu's report, we see that the use of context indeed increases the classification accuracy. However, it is computationally tedious on conventional serial computer. We are therefore looking into the use of the Illiac 4 (via the ARPANET).

PREPROCESSING - The aim of preprocessing is to change the image to a form which is more convenient for information extraction. Most images are degraded by noise and blurring. The reduction of noise and the sharpening of the image generally facilitate information extraction. Several noise reduction techniques are compared by Yoo and Huang. It was found that the

synthetic highs technique works best in balancing the noise level and the edge sharpness.

APPLICATIONS - We are working on several applications, two of which are reported here. Wallace and Wintz present some recent results on using Fourier descriptors for airplane classification. Mitchell and Chen show results of reducing cloud and haze in LANDSAT imagery using three-dimensional digital filters.

APPLICATION OF A FINITE STATE MARKOV PROCESS MODEL TO IMAGE MEASUREMENT

J. Burnett and T. S. Huang

I. Introduction

Our last report [1] briefly mentioned the possibility of using the Viterbi algorithm in conjunction with a finite state Markov process model for making accurate measurements from noisy and blurred images. Here we expand on this idea, derive the algorithm and formulas for calculating the performance of the algorithm.

A Computer simulation of a specific photographic imaging system shows the algorithm to produce asymptotically efficient, and unbiased estimates of object sizes.

Maximum A Posteriori Probability Sequence Estimation

The maximum a-posteriori probability (MAP) estimate of a sequence $\underline{i}$ given a sequence $\underline{z}$ ($\underline{z}$ being a degraded and noisy version of $\underline{i}$) is defined as a sequence $\hat{\underline{i}} = (\hat{i}_1, \hat{i}_2, \ldots, \hat{i}_m)$ such that $P(\underline{i}|\underline{z})_{\underline{i}=\hat{\underline{i}}}$ is a maximum. To calculate $\hat{\underline{i}}$ a model is needed for the relationship between $\underline{i}$ and $\underline{z}$. We assume the observation model shown in Fig. 1. $\underline{i} = (i_1, i_2, \ldots, i_m)$ is the sequence of ideal light intensities with i_k the light intensity of the kth sample point entering the imaging system. Each i_k can assume one of G possible values $a_1, \ldots, a_G$. For example $\underline{i}$ might represent the sequence of reflected light intensities from a scan line of an aerial photo of a bridge across a river. In this case there would be two possible intensity levels: a_1 corresponding to the light reflected from the water and a_2 corresponding to the light reflected from concrete (or whatever construction material was used in the bridge). The state at position k, η_k, is defined to be a set of adjacent intensities $(i_{k-v}, \ldots, i_k, i_{k+v})$. Since each i_j can assume only a finite

number of values each η_k is one of a finite set $[S_1 \ldots S_p]$. Further (to within boundary conditions) there is a one to one correspondence between the state sequence $\underline{\eta}$ and the intensity sequence $\underline{I}$.

The system $h(\cdot)$ represents the degradation of the sequence $\underline{I}$. In the case of photographic imagery this includes blurring due to scattering, diffraction, camera motion, etc. as well as the nonlinear relationship between light intensity and film density. The only assumption that we make on h is that there is a one to one correspondence between $\underline{Y} = (y_1,\ldots,y_m)$ (where $y_k = h(\eta_k)$) and $\underline{\eta}$.

$\underline{N}$ is a sequence of independent noise samples. We do not rule out dependence of the noise parameters on the signal, however. For example film grain noise is approximately normal with a standard deviation proportional to the signal level.

The Algorithm

By definition the MAP sequence estimate $\hat{\underline{I}}$ of $\underline{I}$ is

(1) $P(\underline{I}|\underline{Z})_{\underline{I}=\hat{\underline{I}}}$ is a maximum

but since there is a one to one correspondence between I and $\underline{\eta}$, (1) is equivalent to

(2) $P(\underline{\eta}|\underline{Z})|_{\underline{\eta}=\hat{\underline{\eta}}}$ is a maximum

However,

(3) $P(\underline{\eta}|\underline{Z}) = P(\underline{Z},\underline{\eta})/p(\underline{z}) = P(\underline{Z}|\underline{\eta})\ P(\underline{\eta})/p(\underline{z})$

Due to the Markov assumption of $\underline{\eta}$ and the independence of the noise

$$P(\underline{\eta}) = \prod_{k=1}^{M} P(\eta_{k+1}|\eta_k)$$

$$P(\underline{z}|\underline{\eta}) = P(\underline{z}|\underline{Y}) = \prod_{k=1}^{M} P(z_k|y_k) = \prod_{k=1}^{M} P(z_k|h(\eta_k))$$

Thus we want to maximize

$$(4) \quad \prod_{\ell=1}^{M} P(\eta_{\ell+1}|\eta_1)\, P(z_\ell|h(\eta_\ell))$$

or equivalent minimize

$$(5) \quad \sum_{\ell=1}^{M} - \ln P(\eta_{\ell+1}|\eta_\ell) - \ln P(z_\ell|h(\eta_\ell)) \triangleq \sum_{\ell=1}^{M} \Gamma(\eta_\ell)$$

By assigning a cost or length of $\Gamma(\eta_\ell)$ to each branch of a trellis we can see that the MAP estimate $\hat{\underline{\eta}}$ represents the lowest cost or minimum length path through the trellis.

Let η_1^ℓ be a sequence of states starting at some initial state at position one and ending at state η at position ℓ. In general there will be several possible paths (or sequences η_1^ℓ) through the trellis that pass through state η at position ℓ. Denote this set of paths by Ξ. Let $\hat{\eta}_1^\ell \in \Xi$ be one of these sequences but with the additional restriction that $\Gamma(\hat{\eta}_1^\ell) \triangleq \sum_{j=1}^{\ell} \Gamma(\hat{\eta}_j) < \Gamma(\hat{\eta}_1^\ell)$ for any other sequence $\eta_1^\ell \in \Xi$. (If the two paths have equally low cost any reasonable procedure for deciding between them will do.)

Thus $\hat{\eta}_1^\ell$ (called the survivor sequence or survivor) is the minimum cost sequence or path from the fixed initial state to the state η at position ℓ. Now if the minimum cost complete path from position 1 to position m passes through state η at position ℓ it must have $\hat{\eta}_1^\ell$ as its initial segment (if it did not we would replace the initial segment with $\hat{\eta}_1^\ell$ and get an even lower cost path, a contradiction).

Of course we do not know that the minimum cost complete path passes through state η at position ℓ. However we do know that it must pass through one of p possible states at position ℓ. Thus at any position ℓ we only need store at most p survivor sequences $\hat{\eta}_1^{\ell}$ and their costs $\Gamma(\hat{\eta}_1^{\ell})$. To get to position $\ell+1$ we need only extend all position ℓ survivors by one unit, compute the costs of the possible extensions from

$$(6)\quad \Gamma(\eta_1^{\ell+1}) = \Gamma(\eta_1^{\ell}) + \Gamma(\eta_{\ell+1})$$

and for each of the p possible states $\eta_{\ell+1}$ at position $\ell+1$ select the lowest cost path ending in that state as the position $\ell+1$ survivor. In summary:

Storage: ℓ (position index)

$\hat{\eta}_1^{\ell}$ (p such ℓ point survivor sequences)

$\Gamma(\eta_1^{\ell})$ (costs of each of the p survivor sequences)

Initialization: $\ell=0$

$\hat{\eta}_0^0 = \eta_0$ for each possible initial state

$\Gamma(\eta_0) = -\ell n \pi_{\eta_0}$ where π_{η_0} is the a-priori probability that the initial state is η_0 (if known)

If the a priori probabilities are not known then any reasonable initial cost assignment (such as $\Gamma(\eta_0) = 0$ for all possible initial states η) will do.

Recursion

For each of the p possible states at position $\ell+1$ compute

$$\Gamma(\eta_{\ell+1}, \eta_{\ell}) \triangleq \Gamma(\eta_1^{\ell}) + \Gamma(\eta_{\ell+1})$$

for each possible η_{ℓ} find $\Gamma(\eta_1^{\ell+1}) = \min \Gamma(\eta_{\ell+1}, \eta_{\ell})$ store $\Gamma(\eta_1^{\ell+1})$ and the

corresponding survivor sequence. At position M there will be at most p survivor sequences, one survivor sequence terminating at each permissible position M state. Denote by $\hat{\underline{n}}$ the lowest cost of these sequences. $\hat{\underline{n}}$ is the MAP sequence estimate of $\underline{n}$.

Example. Suppose it is known that at sample point number one the local gray level is zero, that at position eight the local gray level is three and that someplace between these two points a step change between the levels zero and three occurred. The degrading system is linear and shift invariant with impulse response $h_{-1} = h_0 = h_1 = \frac{1}{3}$. The noise is white, Gaussian with variance σ^2. The observed sequence $\underline{z}$ = (-.5, +.25, +.75, .5, 2.8, 2.7, 3.3, 3.1). The possible states are

$S_1 = (0,0,0)$ $\quad$ $S_3 = (0,3,3)$

$S_2 = (0,0,3)$ $\quad$ $S_4 = (3,3,3)$

$h(S_1) = 0$ $\quad$ $h(S_3) = 2$

$h(S_2) = 1$ $\quad$ $h(S_4) = 3$

By assuming that all permissible changes of state are equally likely $-\ln P(n_{k+1}|n_k)$ terms are the same and can be ignored. Thus $\Gamma(n_\ell) = (z_\ell - h(n))^2$. The trellis for this example is shown in Fig. 2. The permissible paths are shown by dashed lines. The numbers are the costs of the survivor sequences to each state. The MAP estimate is shown by a solid line. Its total cost is 1.955 and corresponds to $\hat{i}$ = (0,0,0,0,3,3,3,3).

The minimum cost or minimum length path through a trellis problem and various solution have been around for some time [2]. The algorithm presented above (commonly called the Viterbi algorithm or VA) was presented by Viterbi [9] as a technique for decoding convolutional codes. The algorithm has since

been used by Forney [3] as a solution to the intersymbol interference problem. The algorithm also has applications in text recognition [4] since it can exploit Markov dependence in English text and can be applied to scene analysis problems [5] due to Markov relationships between objects in scenes.

The VA can be used to make measurements of objects in digitized images providing the object whose length or width is to be measured is distinguished from its background by abrupt changes in reflected light intensity at its edges. A MAP sequence estimate of a scan line across the object can be calculated and the edge locations relative to the start of estimated line can be subtracted to produce an estimate of the size.

In the next sections formulas will be derived for predicting the performance of the VA at locating edges and estimating widths.

Error Analysis for Step Edges

In the previous section we presented the VA as a possible technique for making measurements from digitized images. The measurement technique consists of making a MAP estimate of a scan line across the object of interest and subtracting the position of the boundary points of the estimate of the scan line to get an estimate of the width. In this section we examine the accuracy of this technique. In particular we will calculate the probability of mislocating a step edge by $|n|$ $(n = \pm 1, \pm 2, \ldots)$ points. It is assumed that a change in brightness from level a_1 at some initial position to level a_2 at position m has been detected. The observed signal is blurred and noisy so that the location of the step change between levels a_1 and a_2 is uncertain.

Define an error event E_n by the conditions $i_k = \hat{i}_k$ and $i_{k+|n|+1} = \hat{i}_{k+|n|+1}$ but $i_j \neq \hat{i}_j$ for $k+1 \leq j \leq k+|n|$. If p is the number of states then there must $|n| + p - 2$ $(|n| \geq 1)$ state disagreements (see Fig. 3 for p=4, n=-2). If $\hat{\underline{y}}$ is the output vector corresponding to $\hat{\underline{i}}$ and $\underline{y}$ is the vector corresponding

to the true sequence $\underline{I}$ then $\hat{\underline{I}}$ will be decided over $\underline{y}$ if

$$(7)\quad \sum_{j=k-t}^{k+|n|+t} \ln P(z_j|\hat{y}_j) > \sum_{j=k-t}^{k+|n|+t} \ln P(z_j|y_j)$$

where $t = [(P-1)/2]$ = largest integer $\leq (P-1)/2$

If the noise is normally distributed with zero mean and variance σ^2 then (7) becomes

$$(8)\quad \sum_{j=k-t}^{k+|n|+t} (z_j-\hat{y}_j)^2 < \sum_{j=k-t}^{k+|n|+1} (z_j-y_j)$$

or $||\underline{Z}'-\hat{\underline{Y}}'||^2 < ||\underline{Z}'-\underline{Y}'||^2$

where $\underline{Z}' = (z_{k-t},\ldots,z_k,\ldots,z_{k+n+t})$

$\underline{Y}' = (y_{k-t},\ldots,y_{k+t+|n|})$

$\hat{\underline{Y}}' = (\hat{y}_{k-t},\ldots,\hat{y}_{k+t+|n|})$

The vectors $\underline{Z}'$, $\hat{\underline{Y}}'$, $\underline{Y}'$ can be viewed as three points in $|n|+2t+1$ dimensional space that define a plane (see Fig. 5).

Define $\langle \underline{Z},\underline{Y}\rangle = \underline{Z}^T\underline{Y} = \sum_j z_j y_j = \underline{Y}^T\underline{Z}$. Now the condition for deciding $\hat{\underline{Y}}$ over $\underline{Y}$ becomes

$$\frac{1}{||\underline{Y}-\underline{Y}||}\left[\langle \hat{\underline{Y}}-\underline{Z}', \hat{\underline{Y}}-\underline{Z}\rangle < \langle \underline{Y}-\underline{Z}', \underline{Y}-\underline{Z}\rangle\right]$$

(the reason for the normalization by $||\hat{\underline{Y}}-\underline{Y}||$ will be obvious shortly)

$$\frac{1}{||\underline{Y}-\underline{Y}||}\left[\langle \hat{\underline{Y}}, \hat{\underline{Y}}-\underline{Z}\rangle - \langle \underline{Z}', \hat{\underline{Y}}-\underline{Z}\rangle < \langle \underline{Y}', \underline{Y}-\underline{Z}\rangle - \langle \underline{Z}', \underline{Y}-\underline{Z}\rangle\right]$$

$$\frac{1}{||\underline{Y}-\underline{Y}||}\left[\langle \hat{\underline{Y}}, \hat{\underline{Y}}-\underline{Z}\rangle - \langle \underline{Z}', \hat{\underline{Y}}\rangle < \langle \underline{Y}', \underline{Y}-\underline{Z}\rangle - \langle \underline{Z}', \underline{Y}\rangle\right]$$

$$\frac{1}{||\underline{Y}-\underline{\hat{Y}}||}\left[\langle\underline{\hat{Y}}, \underline{\hat{Y}}-\underline{Z}\rangle - \langle\underline{t}, \underline{\hat{Y}}\rangle - \langle\underline{\hat{Y}}, \underline{\hat{Y}}\rangle < \langle\underline{Y}, \underline{Y}-\underline{Z}\rangle - \langle\underline{Z}, \underline{Y}\rangle - \langle\underline{Y}, \underline{\hat{Y}}\rangle\right]$$

$$\frac{1}{||\underline{Y}-\underline{\hat{Y}}||}\left[\langle\underline{\hat{Y}}, \underline{\hat{Y}}-\underline{Z}\rangle - \langle\underline{Y}, \underline{\hat{Y}}-\underline{t}\rangle < \langle\underline{Y}, \underline{Y}-\underline{Z}\rangle - \langle\underline{\hat{Y}}, \underline{Y}-\underline{Z}\rangle\right]$$

$$(9) \qquad \frac{\langle\underline{\hat{Y}}-\underline{Y}, \underline{\hat{Y}}-\underline{Z}\rangle}{||\underline{Y}-\underline{\hat{Y}}||} < \frac{\langle\underline{Y}-\underline{\hat{Y}}, \underline{Y}-\underline{Z}\rangle}{||\underline{Y}-\underline{\hat{Y}}||}$$

Thus an error will occur in deciding between $\underline{Y}$ and $\underline{\hat{Y}}$ whenever the distance from $\underline{Z}$ to $\underline{\hat{Y}}$ along the $\frac{\underline{Y}-\underline{\hat{Y}}}{||\underline{Y}-\underline{\hat{Y}}||}$ axis is less than the distance from $\underline{Z}$ to $\underline{Y}$ along the same axis. Since $\underline{Z}=\underline{Y}+\underline{N}$ the projection of $Z-Y$ on $\frac{\underline{Y}-\underline{\hat{Y}}}{||\underline{Y}-\underline{\hat{Y}}||}$ is determined by the projection of the noise N on the axis. This projection is a linear combination of normal random variables with zero mean and variance σ^2 and hence is also a normal random variable with zero mean and (due to the normalization by $||Y-\hat{Y}||$) variance σ^2. Therefore the probability that $\hat{Y}$ will be decided over the correct path Y is the probability that a normal random variable exceeds half the distance between $\hat{Y}$ and Y. With $T_n = ||Y-\hat{Y}||$ then

$$(10) \quad \text{Prob (decide } \hat{Y} \text{ over } Y) = \int_{\frac{T_n}{2}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-n/2\sigma^2}\, d\eta$$

$$= Q\left(\frac{T_n}{2\sigma}\right).$$

Now $Y - \hat{Y} = (I - \hat{I}) * h$ (assuming a linear and shift invariant degradation)

$$I - \hat{I} = (0,0,0,\Delta,\Delta,\Delta,0,0)$$

where $\Delta = \pm(a_1 - a_2)$

Let $U_1(K)$ be the unit step response of h. Then $||Y - \hat{Y}||^2 = ||Y - \hat{Y}||^2$

$$= \sum_j (y_j - \hat{y}_j)^2 = T_n^2$$

$$T_n^2 = \sum_{j=1}^{M} [(I_j - \hat{I}_j) * h]^2$$

$$= \Delta^2 \sum_{j=k-t}^{k+|n|+t} [U_1(j-K) - U_1(j-|n|-K)]^2$$

$$(11) \quad = \Delta^2 \sum_{j=-t}^{|n|+t} [U_1(j) - U_1(j-|n|)]^2$$

For any value of n there are two possible edge locations estimates that are $|n|$ points in error corresponding to location estimates that are n points before and n points after the correct location. Thus the probability of an error event for step edges with known levels is

$$(12) \quad P(E_n) = 2Q\left(\frac{T_n}{2v}\right)$$

where T_n is given by (11).

Thus far we have only considered the case of constant noise variance. However, for some types of noise such as film grain the variance of the noise varies with the signal.

Consider again the problem of trying to locate a step edge between levels a_1 and a_2 but now assume that the noise variance of the ℓth sample $\sigma^2_{y_\ell}$ depends on y_ℓ. Equation (7) becomes

$$(13) \quad \sum_{j=k-t}^{k+|n|+t} -\ell n\ (\sqrt{2\pi}\ \sigma_{\hat{y}_j}) - (z_j - \hat{y}_j)^2/2\sigma^2_{\hat{y}_j} > \sum_{j=k-t}^{k+|n|+t} -\ell n\ \sqrt{2\pi}\ \sigma_{y_j}$$

$$- (z_j - y_j)^2/2\sigma^2_{y_j}$$

or in an obvious notation

$$(14) \quad ||\underline{Z}' - \underline{Y}'||\,\sigma^2_{\hat{Y}'} < ||\underline{Z}' - \underline{Y}'||\sigma^2_{Y'} + C_{\hat{Y}Y}$$

where $C_{\hat{Y}Y} = \Sigma \ell n \frac{\sigma_{Yj}}{\sigma_{\hat{Y}j}}$.

The exact probability of deciding $\hat{Y}$ over Y can be calculated from (14) by a procedure given in [6] though it is very difficult and will not be considered here. However, one observation should be made. If $\hat{Y}_n$ is the path through the trellis that mislocates the edge by +n points and $\hat{Y}_{-n}$ the path that mislocates the edge by -n points then in general $C_{\hat{Y}_n Y} \neq C_{\hat{Y}_{-n} Y}$ and $||\underline{Z}' - \underline{Y}_n'||\sigma^2_{Y_n} \neq ||\underline{Z}' - \underline{Y}_{-n}'||\sigma^2_{Y_{-n}}$. Therefore the probability that $\hat{\underline{Y}}_n$ is chosen over $\underline{Y}$ will not be the same as the probability that $\hat{\underline{Y}}_{-n}$ is chosen over $\underline{Y}$. This lack of equality between $Pr(n=\alpha)$ and $Pr(n=-\alpha)$ causes the random variable n to have a nonzero mean value which introduces a bias in the edge location estimate. Since $En = \Sigma_\alpha \alpha Pr(n=\alpha)$ and since $Pr(n=\alpha)$ decreases with increasing signal to noise ratio (SNR) the bias will decrease with increasing SNR. Further the decision boundaries (and hence the bias) among the various possible paths are determined by the variances $\sigma^2_{Y_j}$ which in turn are determined by the possible levels a_1, a_2 and the degrading function h. Thus the probability of mislocating the edge by n points (and hence the bias) is independent of the sample point at which the edge occurred.

If the SNR is high enough the bias can probably be ignored. If the SNR is low the bias can be calculated by the procedure mentioned earlier or found experimentally by computer simulation.

Extension to Pulses

In measuring the size of an object two edges must be located. If the object size is sufficiently large or the signal to noise ratio is high enough then with high probability the minimum cost path and the correct path through

the trellis will coincide somewhere inside the object.

With this assumption the events E_1 = first edge mislocated by n_1 points and E_2 = second edge mislocated by n_2 points are independent [7]. Thus if n_w is the number of sample points difference between the true width and the estimated width

(12) $n_w = n_2 - n_1$

$$P(n_w = \alpha) = \sum_{\beta} P(n_2 = \alpha)\ P(n_1 = \alpha - \beta)$$

where $P(n_2 = \alpha)$ and $P(n_1 = \alpha - \beta)$ can be calculated from (10). The sum is over all possible values of β that n_2 can assume such that $n_2 - n_1 = \alpha$.

If the noise varies with the signal level n_w may not have zero mean. As in the previous section the bias can probably be ignored if the SNR is high enough or calculated theoretically or found experimentally if necessary.

Unknown Levels

In the previous section we presented formulas for the probability of mislocating an edge by $|n|$ points. This analysis assumed that the possible levels a_1 and a_2 that of the process were known. In this section the probability of error is calculated assuming that the levels are not known a-priori.

A reasonable course of action in the case of unknown levels is to obtain "training" samples of the gray levels characterizing the object of interest. These samples can be processed in some fashion to produce estimates $\hat{a}_1$ and $\hat{a}_2$ of levels a_1 and a_2, respectively. Suppose $\hat{a}_1 = a_1 + \varepsilon_1$ and $\hat{a}_2 = a_2 + \varepsilon_2$ where ε_1 and ε_2 are normal random variables with zero mean and variance σ_1^2 and σ_2^2, respectively. (If $\hat{a}_1$ and $\hat{a}_2$ are large sample maximum likelihood estimates of a_1 and a_2 then the above model is accurate [8].)

Let the correct path through the trellis by $\underline{Y}$ and the estimate be $\hat{\underline{Y}}$. In this case the correct path is the one that places the edge in the correct position even though the levels (or height) of the step may be incorrect. The "signal space" diagram of Fig. 5 has been repeated in Fig. 6 with the addition of a new point $\underline{\theta}$. $\underline{\theta}$ represents the output sequence corresponding to the true edge location and step levels. In general, $\underline{\theta}$ will not lie in the plane defined by $\underline{Z}$, $\hat{\underline{Y}}$, $\underline{Y}$. However, as before, only the projections of $\underline{Y} - \underline{Z}$ and $\hat{\underline{Y}} - \underline{Z}$ where $\underline{Z} = \underline{\theta} + \underline{N}$ on $\frac{\underline{Y}-\hat{\underline{Y}}}{||\underline{Y}-\hat{\underline{Y}}||}$ will have any effect on the choice between $\hat{\underline{Y}}$ and $\underline{Y}$.

Now $\hat{Y}-Y = \Delta h * (0,0,\ldots,\underbrace{1,1,1}_{n\ 1's},\ldots 0,0)$ $\Delta = \pm(a_1-a_2)$

and $Y-\theta = h * (\varepsilon_1,\varepsilon_1,\ldots,\varepsilon_1,\varepsilon_2,\varepsilon_2,\ldots,\varepsilon_2)$. Thus the projection T' of $Y-\theta$ on $\frac{\underline{Y}-\underline{Y}}{||\underline{Y}-\underline{Y}||}$ will be a linear combination of ε_1 and ε_2. Since a linear combination of normal r.v's is again normal the only information needed to characterize T' is its variance (it will have zero mean since ε_1 and ε_2 were assumed to have zero mean) σ_t^2 which can be calculated from knowledge of σ_1^2, σ_2^2, n_1, and h.

Again, by arguments similar to those in the previous sections the probability of deciding $\hat{Y}$ over Y is the probability that the noise along the $\hat{Y}$-Y axis exceeds $\frac{T_n}{2}$ - T' or equivalently probability that the sum of two normal random variables with variances σ^2 and σ_t^2 exceeds $\frac{T_n}{2}$

$$(13) \quad = Q\left(\frac{T_n}{2(\sigma^2 + \sigma_t^2)}\right) \quad \text{and}$$

$$P(E_n) = 2Q\left(\frac{T_n}{2(\sigma_n^2 + \sigma_t^2)}\right)$$

This result can be extended to pulse width accuracy as before. If $n_w = n_1 - n_2$ is the number of sample points of error in the width the estimate then

$$Pr[n_w=\alpha] = \sum_{\beta} Pr[n_1=\beta]\ Pr(n_2=\alpha-\beta)$$

where $Pr(n_1=\beta) = Q(\frac{T_\beta}{2(\sigma^2+\sigma_t^2)})$. Equation (13) shows that if the estimates $\hat{a}_1$ and $\hat{a}_2$ are such that $\sigma_t^2 << \sigma^2$ then the lack of a-priori knowledge about the possible levels will have very little effect on the probability of mislocating an edge.

Simulation Results

A pulse of width thrity sample points was generated and blurred by a linear shift-invariant system with a Gaussian shaped impulse response with a standard deviation of one sample point. This blurred pulse was transformed by $y_k = 1.066\ (\log i_k - 1.5.) + .2$ where i_k is the $k^{\underline{th}}$ sample of the blurred pulse. This transformation simulates the d-log E curve of film. Noise of standard deviation $.4Yk^{1/3}$ was added. The noisy blurred signal was then processed to produce an estimate of the pulse width. Several different SNR's were used with one hundred width estimates obtained at each SNR. The results are shown in Figs. 6 and 7. The bias in the width estimate can be seen to decrease with increasing SNR as expected. The variance of the estimate can be seen to decrease with increasing SNR and appears to asymptotically approach the Cramer-Rao lower bound.

REFERENCES

[1] J. Burnett and T. S. Huang, "Image Decomposition," in Image Analysis and Modeling, Interim Report, 1975.

[2] M. Pollack and W. Wiebenson, "Solutions of the Shortest Route Problem - A Review," Oper. Res., Vol. 8, Mar. 1960.

[3] G. Forney, "Maximum Likelihood Sequence Estimation of Digital Sequences In the Presence of Intersymbol Interference," IEEE Trans. Inf. Theory, Vol. IT-18, May 1972.

[4] J. Raiviv, "Decision Making in Markov Chains Applied to the Problem of Pattern Recognition," IEEE Trans. Inf. Theory, Vol. IT-13, Oct. 1967.

[5] F. Preparata and S. Ray, "An Approach to Artificial Nonsymbolic Cognition," Inform. Sci., Vol. 4, Jan. 1972.

[6] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, New York, pp. 62-65, 1972.

[7] G. Forney, "The Viterbi Algorithm," Proc. IEEE, Vol. 61, March 1973.

[8] H. Van Trees, Detection, Estimation, and Modulation, Part I, Wiley, New York, p. 71. 1968.

[9] A. Viterbi, "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," IEEE Trans. Inf. Theory, Vol. IT-13, April 1967.

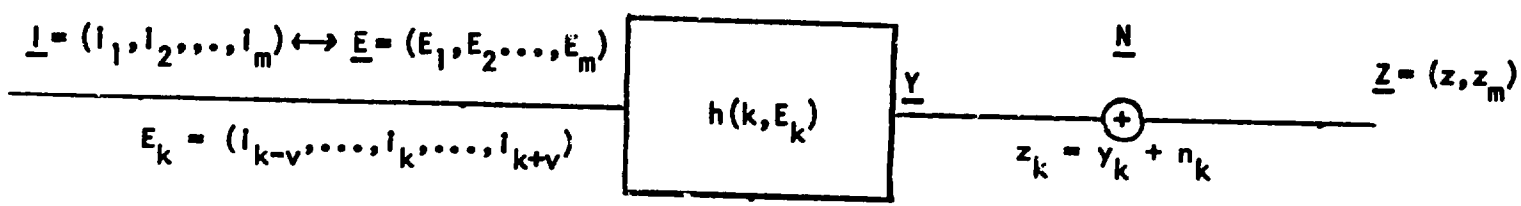
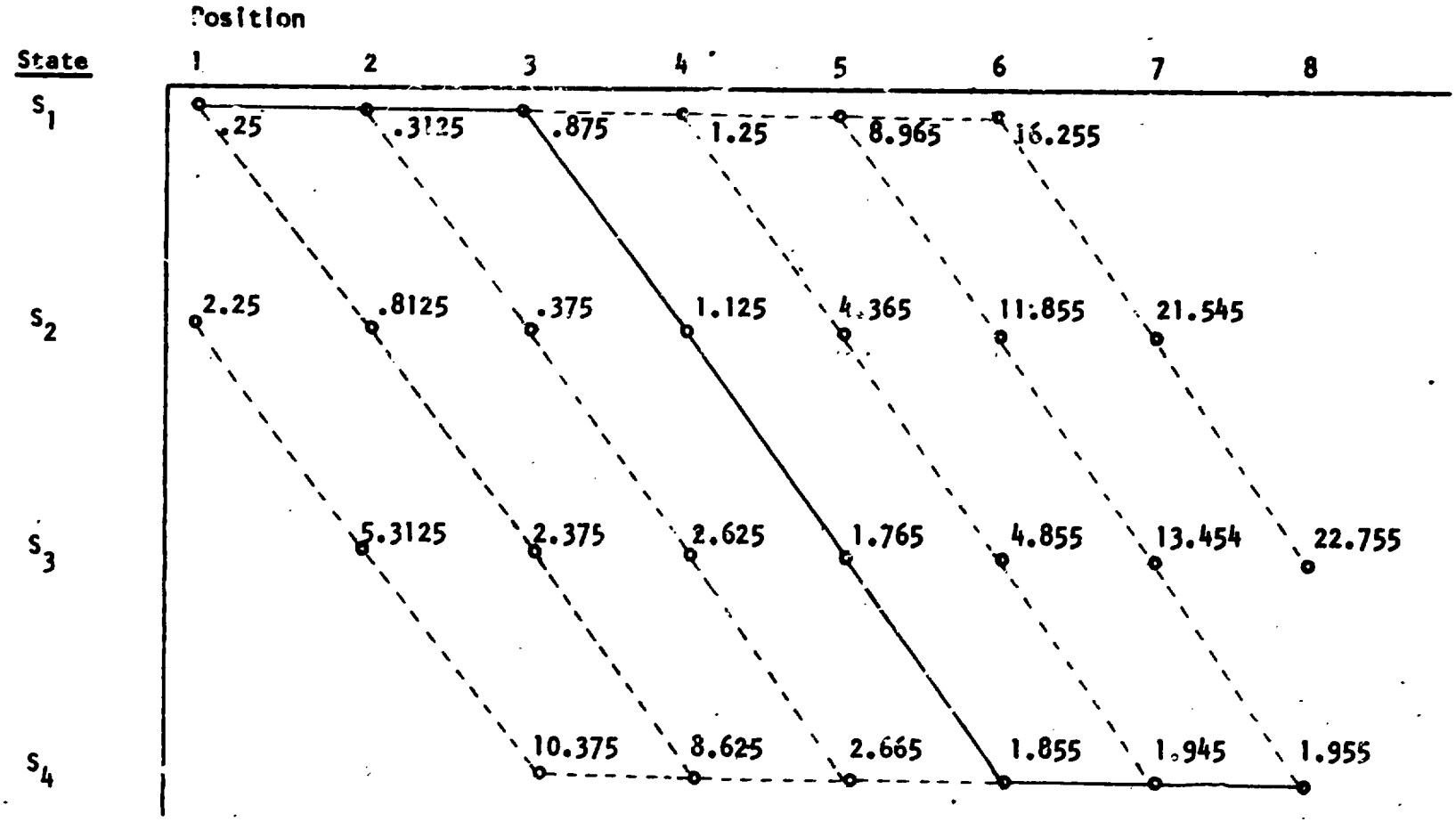


Figure 1 Observation model



$\hat{\underline{E}} = (S_1, S_1, S_1, S_2, S_3, S_4, S_4, S_4) \leftrightarrow \hat{i} = (0,0,0,0,3,3,3,3)$

Figure 2 Trellis for example 1

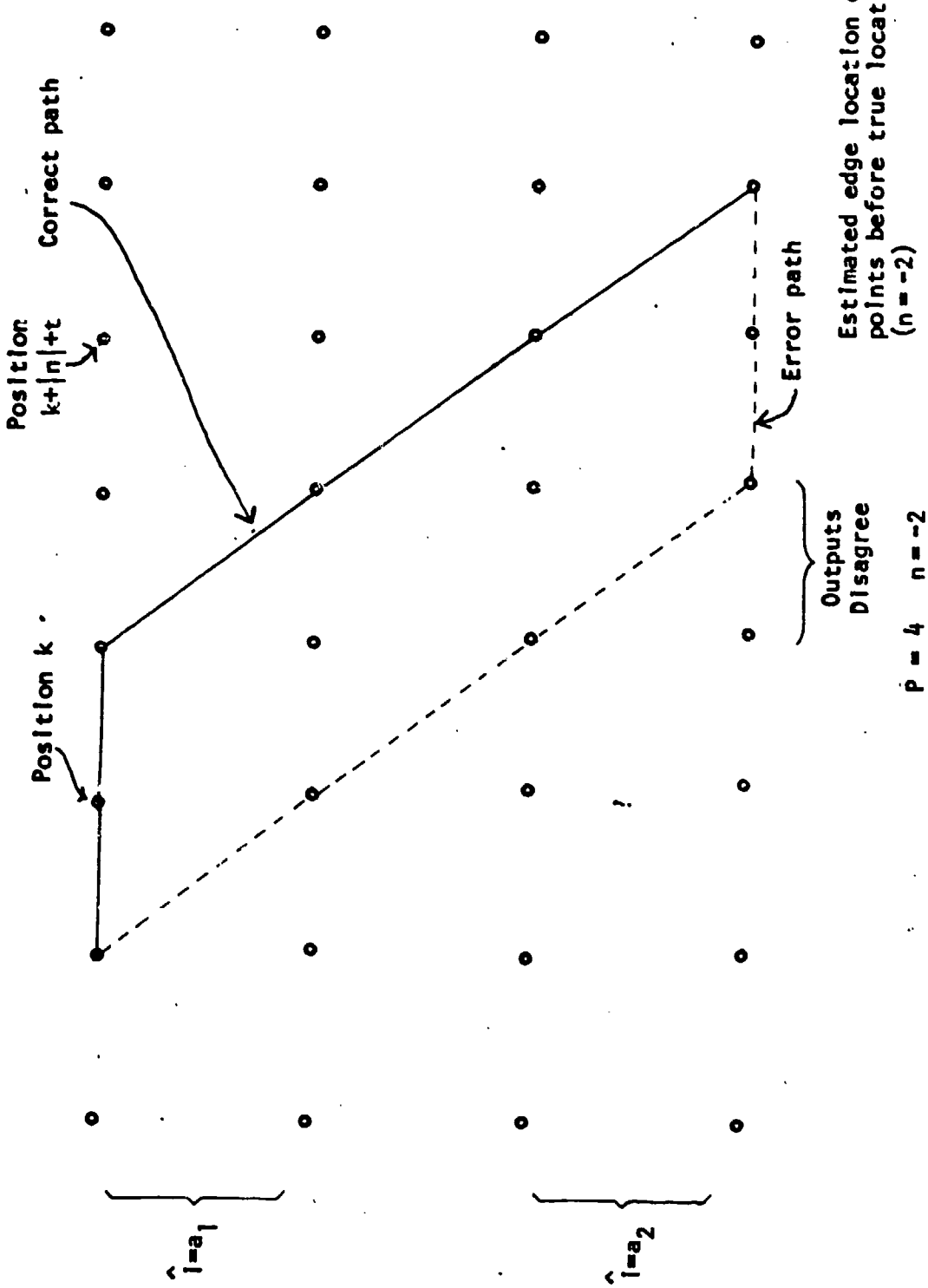


Figure 3 Trellis for an error event

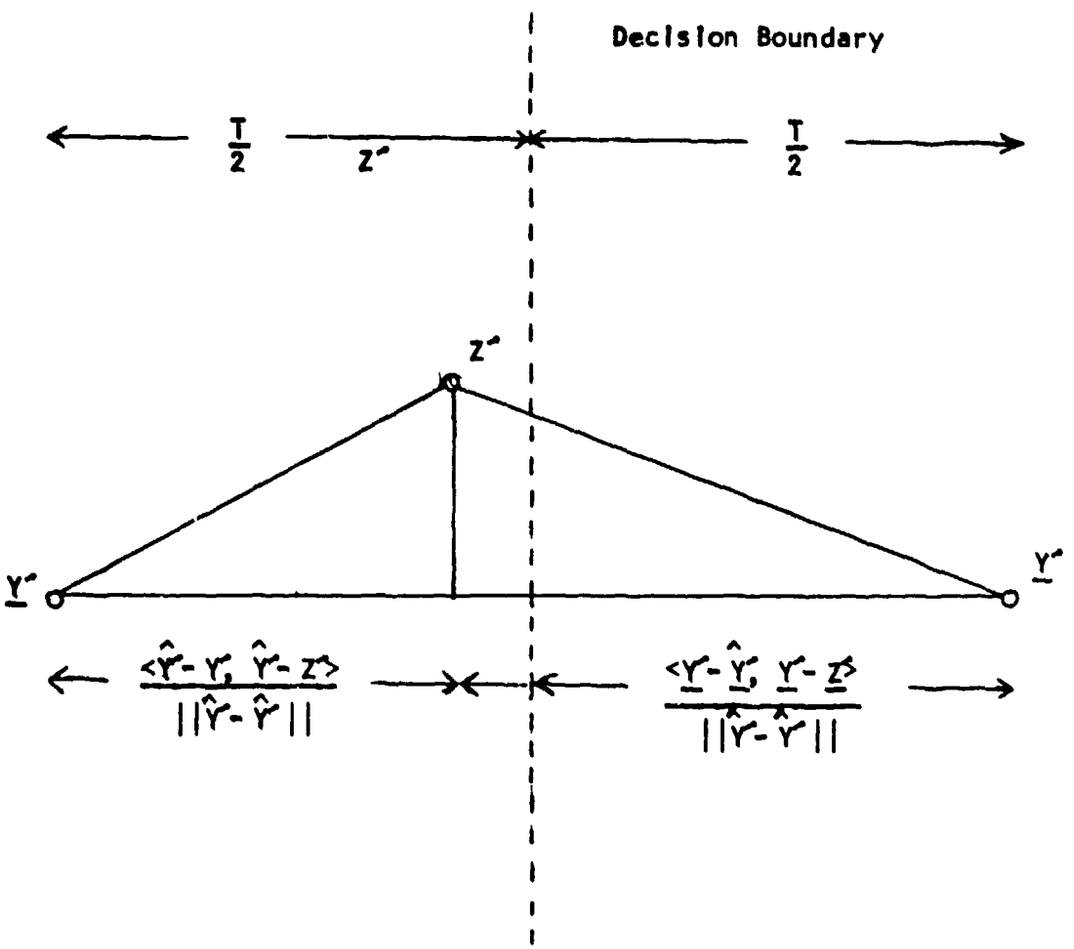


Figure 4 Probability of error calculation

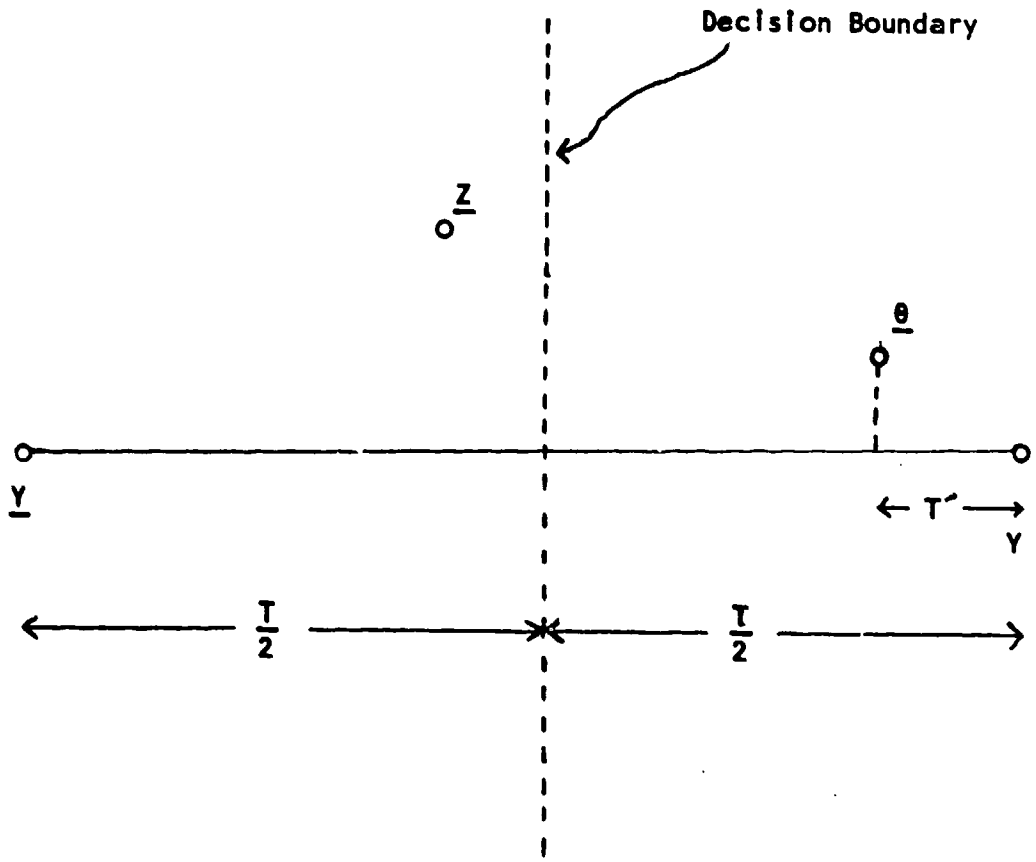


Figure 5 Probability of error calculation for unknown levels

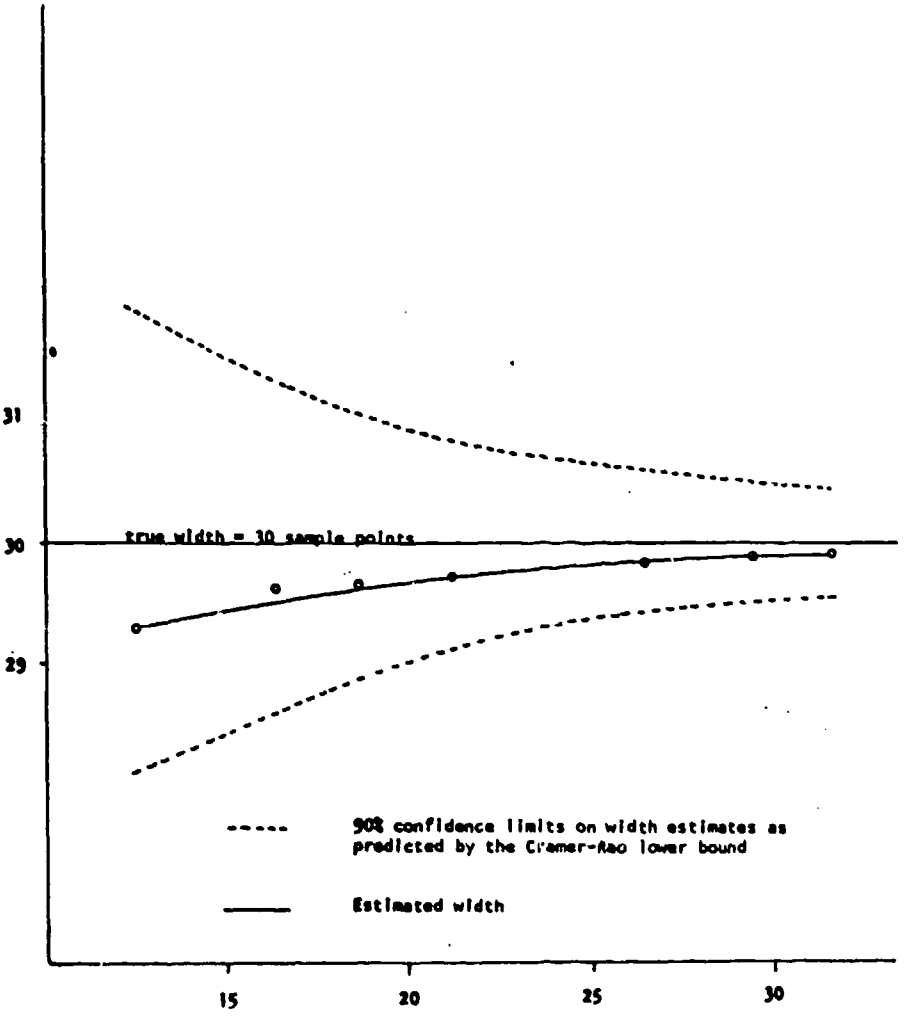


Figure 6 Accuracy of the VA vs. SNR (in dB)

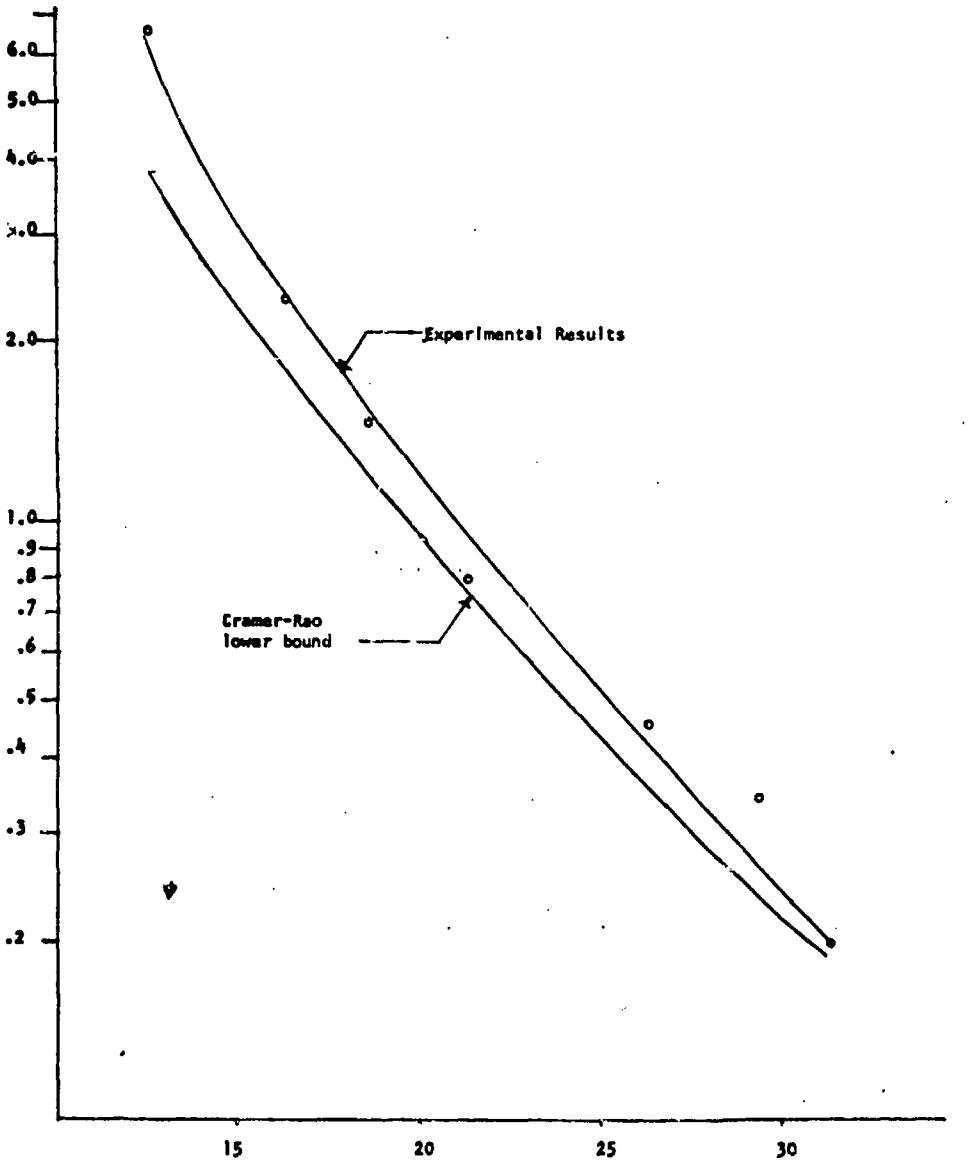


Figure 7 Variance of the width estimate vs. SNR

DIGITAL STRAIGHT EDGES

G.Y. Tang and T.S. Huang

Studies of the digitization of a stright line have been made by Rosenfeld [1], Freeman [2], Morse [3], Gaafar [4], Brons [5]. A set of rules has been established to govern the digitization of an ideal straight line which can be described by a first order mathematical equation. But, in the real world, most edges which appear to be straight to our eyes do not fall into this category. On the contrary, they suffered from noise, degradation, blurring, etc. In this report, we are going to show some experimental results on real straight edges and to compare them with the ideal case. A simple testing algorithm is then developed to determine if a real edge is a straight one or not. The application of the testing algorithm can be found in various areas such as syntactic pattern recognition, character recognition, scene analysis and efficient contour coding, etc. Our ultimate goal for this study is to use syntactic method to aid us to detect edges in a noisy environment.

In the following, we are to use the word 'real' to refer to whatever is on pictures obtained by practical imaging devices. The word 'ideal' refers to the ideal case.

I. PROPERTIES OF AN IDEAL STRAIGHT LINE:

An intensive study of the properties that a digitization of a straight line should have has been reported previously. A brief summary is given here.

(A) The Chord property, proposed by Rosenfeld, is a necessary and sufficient condition for a digital arc to be straight.

(B) The digitization of any real curve can be expressed in terms of a chain code [2]. The chain code of a straight line should obey the following rules.

1) There are at most two different code elements differing by 1, modulo 8.

2) One of these two codes occurs singly.

3) Successive occurrances of the single element are as uniformly spaced as possible.

C) For an ideal straight line with rational slope, the corresponding chain code is a repetition of a certain period.

D) We can always find a rational slope to approximate a real slope so that their chain codes are as close to each other as we want.

E) An ad hoc testing algorithm has been proposed [3] to see if a given chain code may come from a straight line.

II. EXPERIMENT

The nice properties listed in the foregoing section are based on the assumption that the lines are ideally straight, the digitization is noise-free and the digitization scheme is well defined [1], [2]. In the real world, where a model with such a high order of idealism can hardly find many applications, minor randomness corrupts the structures severely. The intention of our experiment is to observe the code structures of real edges and to compare them to the ideal case. Also we outline a new testing algorithm from our experimental results.

A set of 12 pictures has been taken by a standard Nikon F-2 camera with 35 mm film. A flying-spot scanner is used to discretize the picture into a square matrix of size 256 by 256. The grey levels range from 0 to 255. A simple contour follower and a chain code encoder is used to obtain the chain codes on each picture. Only straight edges are on these 12 pictures. There are totally 30 straight edges.

A contour follower is designed so that the output of the contour follower is the chain codes of the edge which it follows. The input to the contour follower is a window of size 3x3. We assume that the center of the window is

on the contour and that we know which of its eight neighbors is also on the contour. The task of the contour follower is to find a point which is supposed to be the next contour point among the eight neighbors around the center. Then the window moves to and centers at that point. The direction of move is encoded by Freeman's method. Repeating the same procedure, we can obtain the whole contour. The way to locate the next point from the eight neighbors of the center is simply looking at the eight neighbors in counter-clockwise sense starting with the neighbor already known on the contour. A thresholding technique is used to discriminate the two regions defining the contour.

Table 1 shows which ideal properties are violated by chains obtained from real pictures.

The ideal edge is not what one finds in the images produced by real-life imaging devices. There are several factors that degrade the edges that are actually found. Two predominant ones are blurring, or defocusing, and irregularities of the surface structure of the object. Besides, for the case of straight edges, a slight concavity or convexity can be considered as another kind of disturbance. The net effect of these disturbances of Freeman chain code can be summarized as:

1) A third code element is introduced if there is a missing or spurious lattice point. This third code element will occur together with the code element which occurs singly. Figure 1 illustrates this.

2) A third run length is introduced if the missing or spurious lattice point occurred at the beginning or at the end of each run. Fig. 2 illustrates this.

3) Other types of errors occur which cannot be characterized easily. Fig. 3 shows us the test pictures.

III. STRAIGHT LINE RECOGNITION

To recognize a real chain code as a straight line is not straightforward since we have no rigorous definition of what is a real straight edge. Here we propose to employ a heuristic method to solve the problem of the recognition of digital straight lines.

The basic strategy of our approach is rather simple. We are attempting to enclose a digital straight line by a straight band. If such a band with reasonable bandwidth could be found for a given chain code, then we claim that that chain code corresponds to a straight line. Mathematically, a straight band is defined by an equality

$$|Y - mX| \leq k$$

where m is the slope of the straight band and k is related to the bandwidth of the straight band. Two parameters, m and k, are therefore to be determined from a given chain code. We proceed as follows:

Step 1 = Check if there are only two code elements; one occurs singly.

Yes; go to Step 3

No; go to Step 2.

Step 2 = Correct all possible first kind of errors mentioned in the previous section. Then check if there are only two code elements left; one occurs singly. If it is yes, then go to Step 3; otherwise do the following:

Calculate the percentage of the 3rd, 4th ... etc. code elements to the total length of the chain code and the percentage of the second code elements which should occur singly in ideal case but occurs in runs in this case to the total number of the second code elements. Then compare these two percentages to two preset thresholds. Once they are smaller than the thresholds, go to Step 3; otherwise the chain code is rejected as a straight line.

Step 3 = m is the ratio of the element which occurs singly to the total number of 1st and 2nd elements.

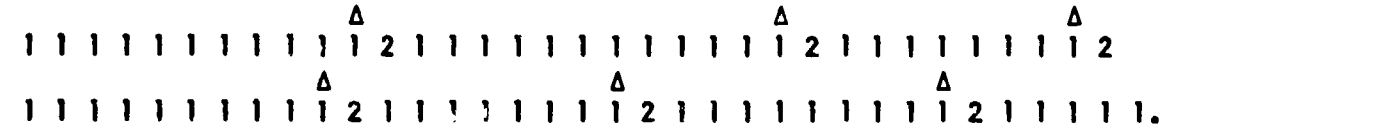
k is a subjective quantity. It relates to the noise level of the picture, the sharpness of the edge and the resolution of the picture.

It seems that so far we have taken few advantages of the nice structural regularities existed among ideal straight lines to aid us in developing testing algorithm for the non-ideal case. A rather straightforward heuristic method has been devised to further reduce the number of testing points. It is noted that the straight band is a geometrically convex set. So, only the two ends of each run on the chain codes need to be tested. Furthermore, if we apply a simple operation which is to replace each run by its run length and to delete the code elements occurred singly, then the output of the operation should obey the rule: only two code elements; one occurs singly, for the ideal case. For the real chain code, we may apply the same operation iteratively to it until the output fails to follow the rule. We then use the final one as a clue to break the initial input chain into several pieces of line segments. Only the ends of each line segment are to be tested. An example illustrating this follows:

A chain code from picture No. 7 in our experiment is

0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1
0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0
1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1
0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1.

The first output of our operation is:



The next output is:

11 12 8 10 8 9 5 . It fails to follow the rule. So we trace back and locate the potential testing points (with Δ) on each generation. The coordinates of the testing points with respect to the initium are then (22,11), (49,24), (68,33), (90,44), (110,53), (131,63).

Table 2 shows the result of applying the foregoing method to test 17 chain codes obtained in our experiment. The "MIN THRESHOLD" means the smallest k value we have to choose in order to report the edges straight. Also we applied Morse's algorithm to test these real chain code. It does not work as well.

REFERENCES

[1]. A. Rosenfeld, "Digital Straight Line Segments," IEEE Trans. on Computers, Vol. C-23, No. 12, Dec. 1974.

[2]. H. Freeman, "Boundary Encoding and Processing," in Picture Processing and Psychopictronics, B.S. Lipkin and A. Rosenfeld, Ed. New York, Academic, 1970, pp. 241-266.

[3]. S. P. Morse, "Computer Storage of Contour-Map Data," Proc. 1968 ACM NAT. Conf., 1968, pp. 45-51.

[4]. M. Gaafar, "Convexity Verification, Blockchords, and Digital Straight Line," (to appear).

[5]. R. Brons, "Linguistic Methods for the Description of a Straight Line on a Girl," Computer Graphics and Image Processing (1974) 3, (48-62).

Line No.	1	2	3	4	5	6
2-1	X	X	X	X	X	X
2-2	X	X	X	X	X	X
2-3	X	X	X	X	X	X
2-4	X	X	X	X	X	X
3-1	V	V	V	V	V	X
3-2	V	V	X	X	X	X
3-3	V	V	X	X	X	X
3-4	V	V	X	X	X	X
4-1	V	V	X	X	X	X
4-2	V	V	V	V	V	X
4-3	V	V	V	V	X	X
4-4	V	V	V	V	V	X
4-5	V	V	X	X	X	X
4-6	V	V	X	X	X	X
4-7	V	V	V	V	V	V
4-8	V	V	V	V	V	V
5	V	V	X	X	X	X
6	X	X	X	X	X	X
7	V	V	V	V	V	X
8	V	V	X	X	X	X
9-1	V	V	V	V	X	X
9-2	V	V	X	X	X	X
10-1	V	X	X	X	X	X
10-2	V	V	X	X	X	X
11-1	V	V	X	X	X	X
11-2	V	V	X	X	X	X
12	V	V	X	X	X	X
1-1	X	X	X	X	X	X
1-2	V	V	V	V	V	V
1-3	X	X	X	X	X	X

1. Only two code elements
2. One of the two code elements occurs singly
3. Only two run lengths
4. Two run lengths are consecutive integers
5. One of the two run lengths occurs singly.
6. The runs of the run lengths can be of at most two run lengths: one of which occurs singly

Table 1. Comparison of real cases and ideal case.

Line No.	Length	Min. Threshold	MORSE
3-1	147	.795	NO
3-2	49	.755	NO
3-3	54	.833	NO
3-4	58	.724	NO
4-1	82	1.329	NO
4-2	47	.404	YES
4-3	42	1.524	NO
4-4	127	1.055	NO
5	238	1.345	NO
6	237	1.456	NO
7	130	.435	YES
8	209	1.004	NO
9-1	85	1.17	NO
9-2	245	.942	NO
10-1	147	1.258	NO
10-2	40	.900	NO
11-2	181	1.127	NO

Table 2 The result of the proposed method in comparison with Morse algorithm.

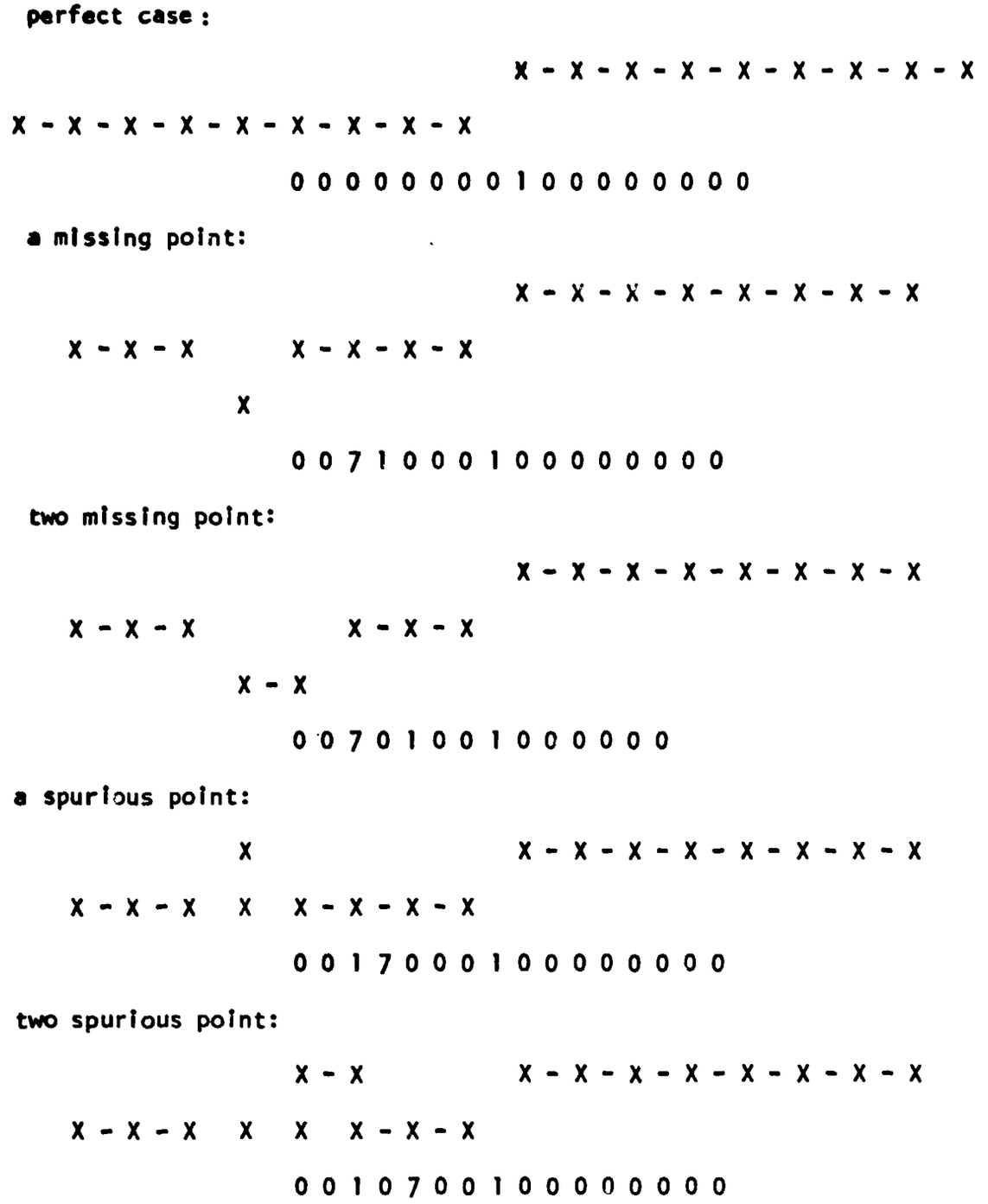


Fig. 1. Missing or spurious points occurred in runs

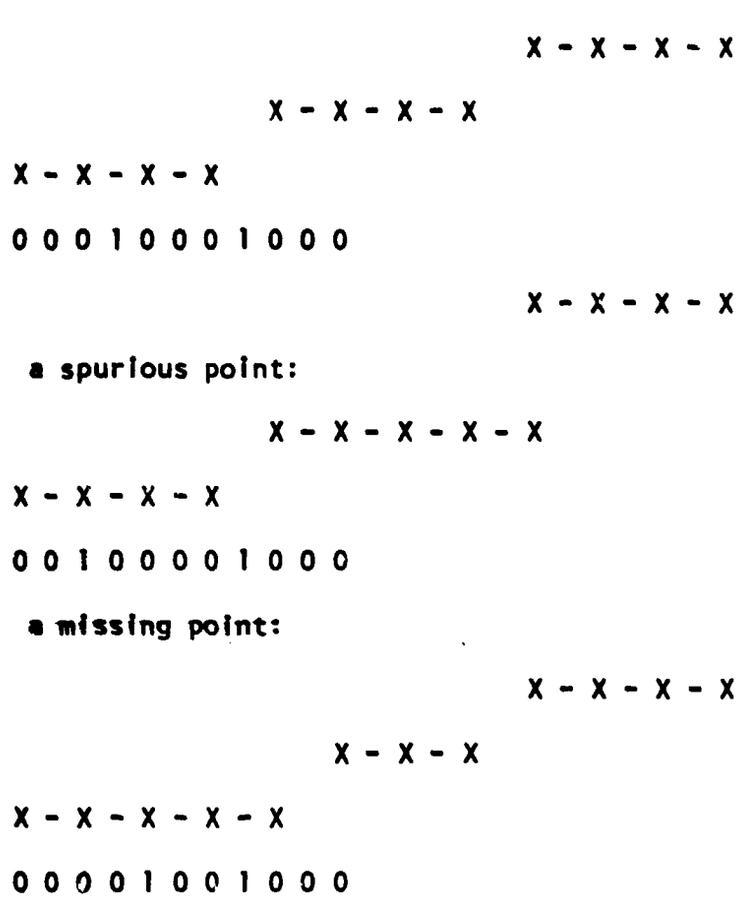
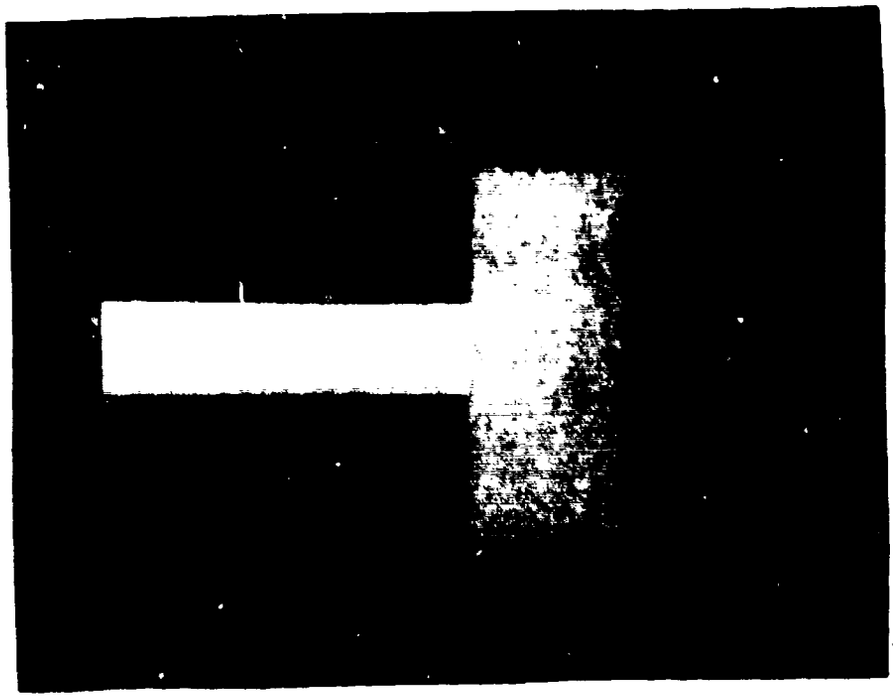
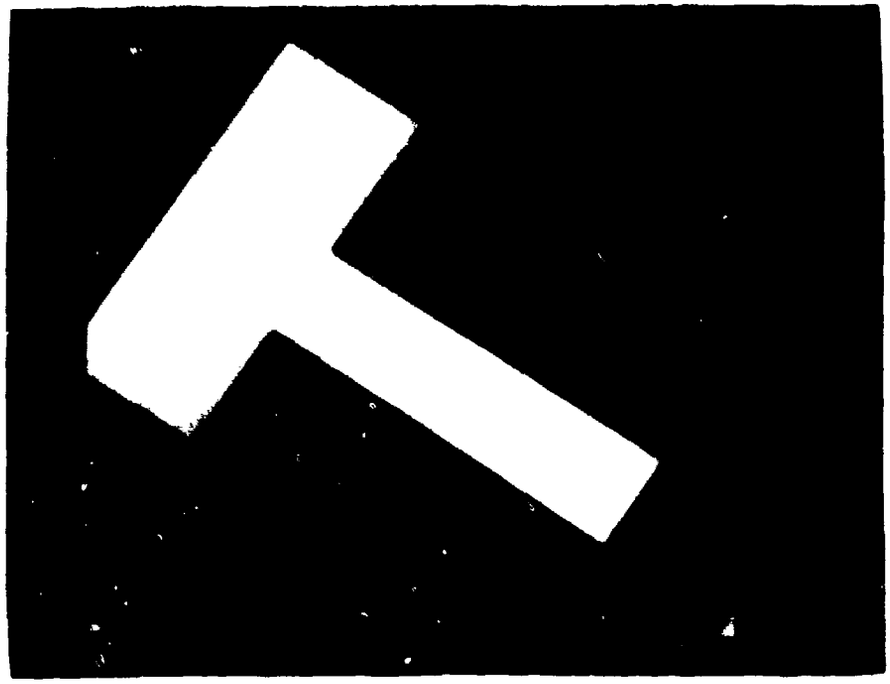


Fig. 2 Missing or spurious points occured at the ends of a run

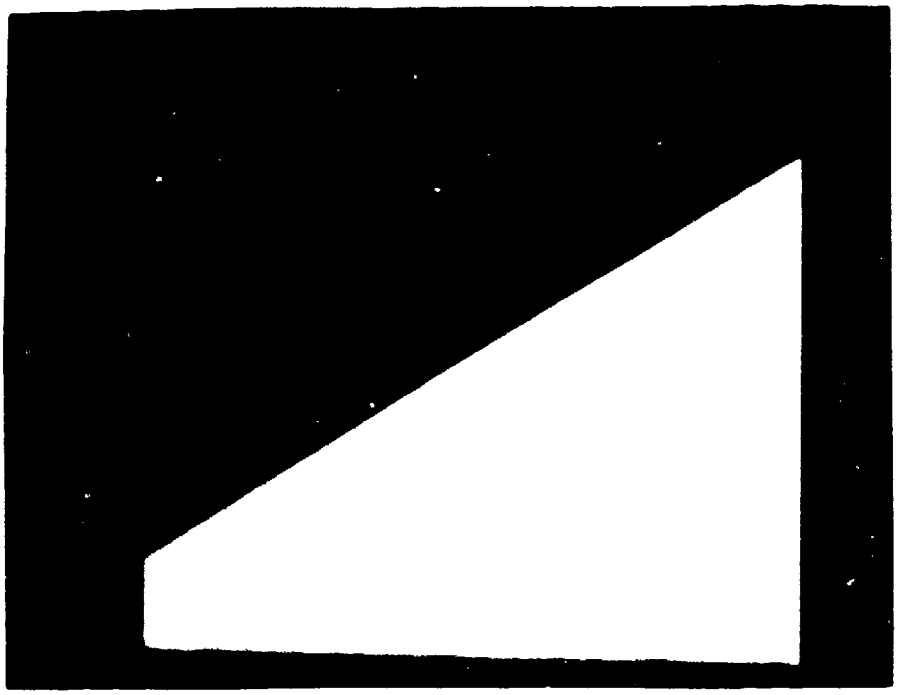


No. 2

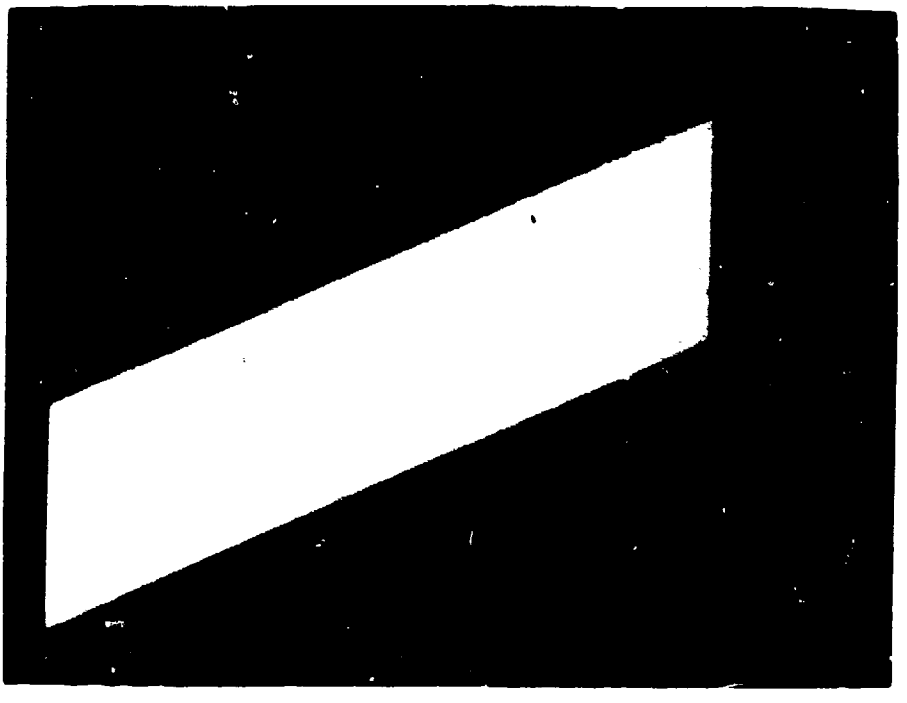


No. 4

Figure 3 Examples of test pictures



No. 5



No. 7

Figure 3 Examples of test pictures

IMAGE STRUCTURE: SYNTACTIC SCENE ANALYSIS

K. S. Fu and R. Y. Li

The essential problem in our research on the syntactic pattern recognition of highways and rivers is really to find a grammar that will describe well these classes of interest. If the physical shape of the class under consideration is completely known and fixed, like printed English character; we can immediately write down the syntactic rules to describe its structure. Since this is not quite true in our case, the construction of grammatical rules has to be based on informations obtained from a set of sample patterns known to come from that class. Hopefully, this set of inferred rules should be able to describe and predict other sample patterns which are of the similar nature as the original training samples and presumably in the same class. A basic approach of grammatical inferrence problem is to construct a grammar by identifying the syntactic structures of the known string and any possible recursiveness that might happen. There are three steps:

(1) Try to discover the syntactic structure of the given string by looking for repetition and dependent relationships.

(2) Decide what sublanguages make up the language and generate non-terminals for each sublanguages.

(3) Combine equivalent nonterminals which have almost the same sublanguage and determine the appropriate relationships among sublanguages.

One practical method to learn the syntactic structures of the given pictures is to use a semantic teacher to learn the meaningful nonterminals one level at a time [1]. To start the inferrence process, we first find the types of terminals or primitives that will fit the subparts of the picture pattern for a given window size. After this initial extraction process, we have to decide the most probable combinations of primitives which occur as neighbors of each

other in the set of observed training samples. These combinations are then applied to the training data set to test their recognition effectiveness. When the results appear to be satisfactory after some additions and deletions of the combination rules, we can choose this set of rules to represent the training samples. The appropriate grammar can then be formulated by using these rules. In our present case, we choose the tree grammar because of its easiness to describe these rules. A tree recognition program based on this tree grammar can then be used to recognize the training data set of Lafayette and a test data set, that of Grand Rapids, Michigan. Figures (1), (2), and (3) contain the results from Lafayette experiment. Other preliminary results from Grand Rapids can be found in reference [3].

References

[1] Brayer, J.M. and K.S. Fu, "Web grammar and its application to pattern recognition," TR-EE 75-1, Purdue University, W. Lafayette, In 1975.

[2] Evans, T.G., "Grammatical inferrence techniques in pattern analysis", Software Engineering, Vol. 2, T.J. Tou (ed.), Academic Press, 1971.

[3] LI, R.Y. and K.S. Fu, "Tree system approach for LANDSAT data interpretation", Purdue Symposium on machine processing of remotely-sensed data, June 29 - July 1, 1976.

Figure (1) Statistically-classified results of Lafayette

Figure (2) Skeltons from Lafayette samples

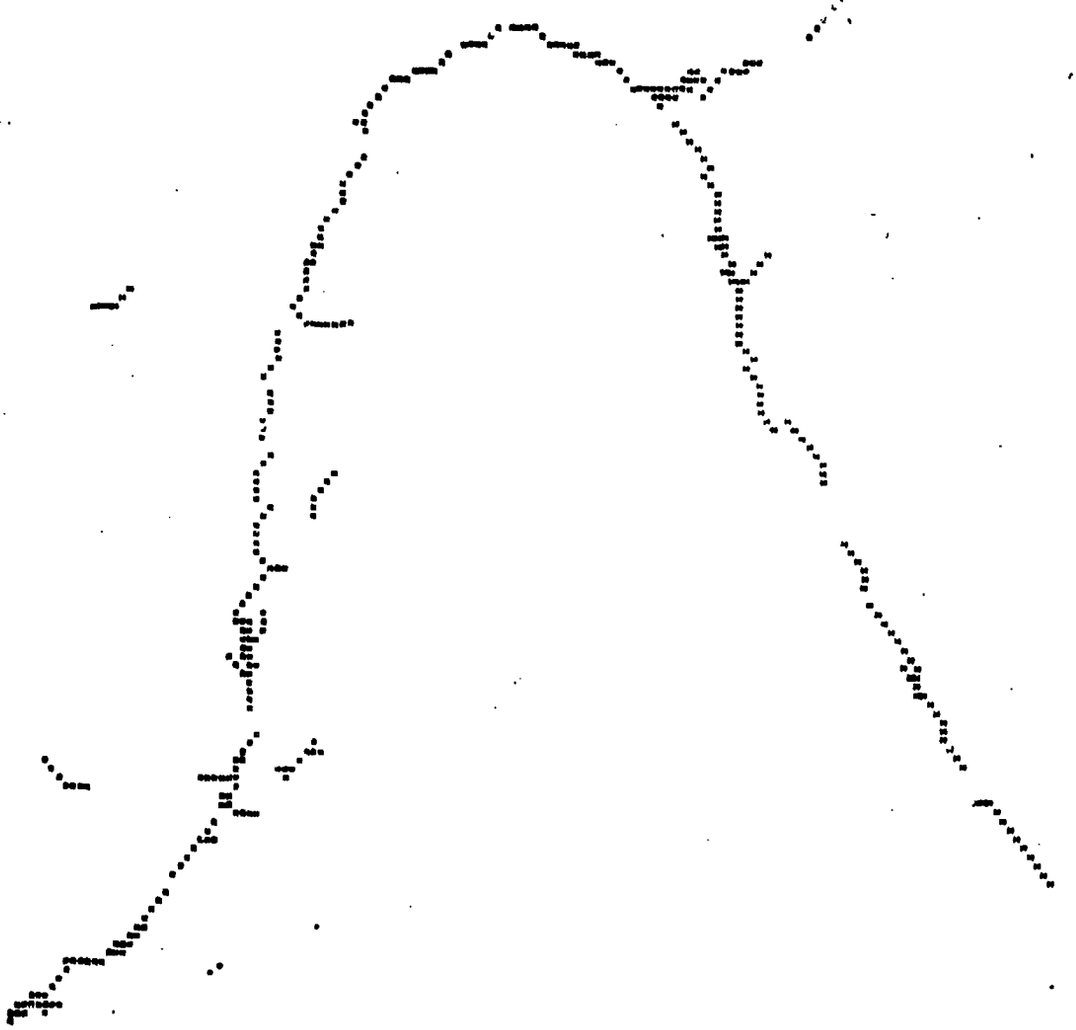


Figure (3) Syntactically-recognized results

SYNTACTIC SCENE SEGMENTATION

K. S. Fu and J. Keng

A syntactic approach to scene segmentation has been investigated which involves two levels of processing. The first level, referred to as the transformation process, consists of five steps referred to as (1) threshold finding, (2) horizontal processing, (3) vertical processing, (4) logic integrating, and (5) line smoothing. The second level, which is the actual syntactic analysis, requires inference of a tree grammar to describe the boundaries of homogeneous regions. The tree grammar is then implemented in a parser which traces the region boundaries.

The approach has been implemented and initial experiments on multispectral remote sensing imagery are being conducted. Further detail and experimental results will follow.

Working on the problem of picture segmentation through a syntactic approach, we feel that the evaluation of the earth resources is very useful. So, the multispectral remotely sensed picture is chosen as data.

There are two levels of processing for the syntactic picture segmentation, first, the transformation process and, second, the tree grammar analysis. The first level consists of five sub-processes, threshold finding, horizontal processing, vertical processing, logic integrating, line smoothing. The process of tree grammar analysis utilizes the corresponding parser from the inferred tree grammars to process the transformed picture. Then a picture is segmented.

Syntactic Picture Segmentation

a. First level - transformation processing

(1) Threshold finding

From a digitized picture, training regions are located from every homogeneous parts of the picture, then the differences between two of these means of grey levels of the training regions are calculated. If they are close to zero we neglect them, then the minimum of them is selected as a threshold.

(2) Horizontal processing

The model of the multisectral images is defined on the Euclidean n-dimensional space E^n which is a space having n coordinates $x_1, x_2, \ldots, x_n$. The number n here represents the number of channels to be chosen. A point of the space is by definition an ordered n-tuple $(x_1, x_2, \ldots, x_n)$. The distance between two points is defined as Euclidean distance. The operation procedure is to compare the grey levels of (1,1) and (1,2). If the distance is smaller than threshold, then set zeroes to (1,1) and (1,2). Then, (1,1) and (1,3) are compared, if the distance is greater than threshold. A one is put to (1,3) and the same operations start from (1,4). The operations on second row follow the same pattern.

(3) Vertical processing

The operations are the same as horizontal processing except it goes vertically instead of horizontally.

(4) Logic integrating

The logic variable of horizontal processing is named as H and V for the vertical processing. The logic integrating process achieves the integration through Boolean algebra V+H.

(5) Line smoothing

The line smoothing algorithm connects the discontinuity of the lines. As for this part, lots of line smoothing algorithms can be devised.

b. Second level - tree grammar analysis

A tree grammar is inferred to describe the boundaries of the homogeneous regions. The tree grammar is used to trace the boundaries and reject the unnecessary boundary parts. Finally a segmented picture is received from the (two level) syntactic method.

The scheme of syntactic picture segmentation has been implemented and the experiments have been conducted on the multispectral remotely sensed pictures of an area in the State of Indiana. The pictures were taken on August 13, 1971 and stored in the computer IBM 360/67 of the Laboratory for Applications of Remote Sensing in West Lafayette, Indiana.

The result of a picture 96x96 area is shown in Fig. 1. For the purpose of comparing processing time and accuracy, a result of statistical segmentation by clustering is also provided in Fig. 2. The computer processing time on IBM 360/67 for the same area of the picture, the syntactic picture segmentation takes about 36 seconds and the clustering technique requires 180 seconds.

The area of column (105-132) and row (444-450) in Fig. 1, shows lots of boundary points. From a survey of the ground truth (Fig. 3), it points out the reason. Because this area is pasture and bare soil compound area. So lots of boundary between these two objects of the compound area are located. A classified result of the same area is also provided in Fig. 4. In the region column (90-100), row (480-488) in the syntactic segmentation result

shows a segment which corresponds to the same location in the classified result, But the clustering result can not segment it well.

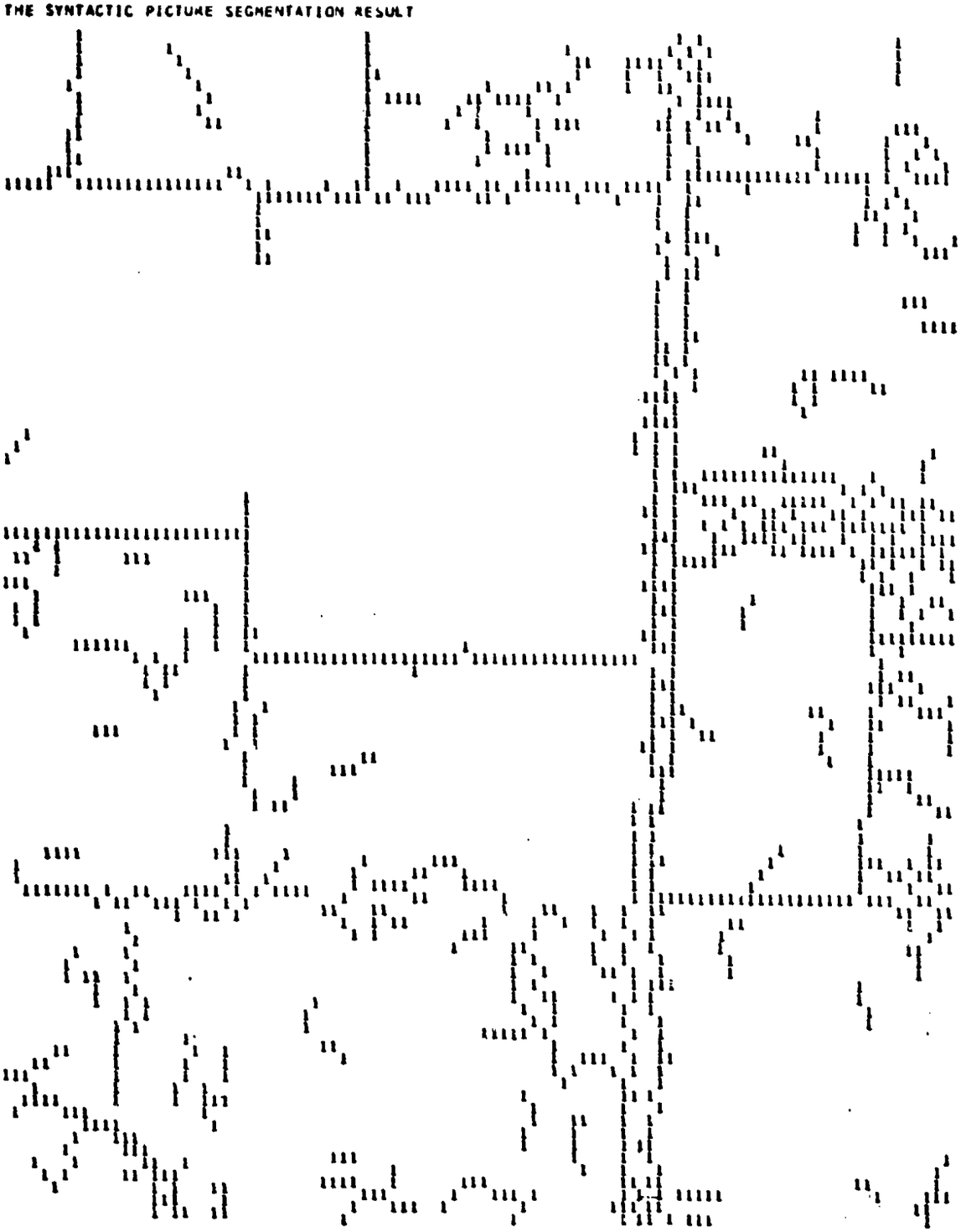


Figure 1 The syntactic picture segmentation result

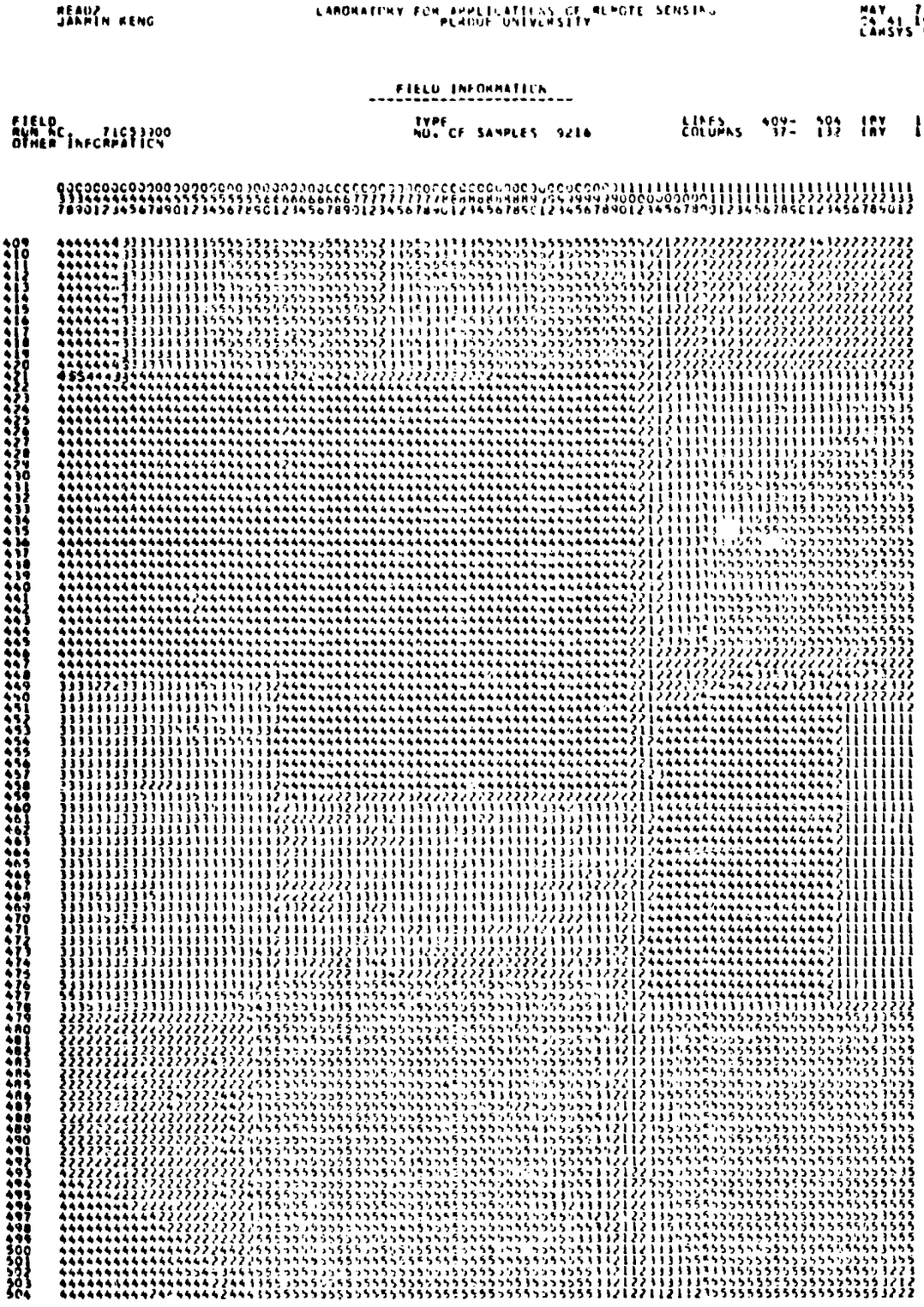


Figure 2 The statistical picture segmentation (clustering) on same area as Fig. 1

Figure 3 The ground truth of the area of Fig. 1

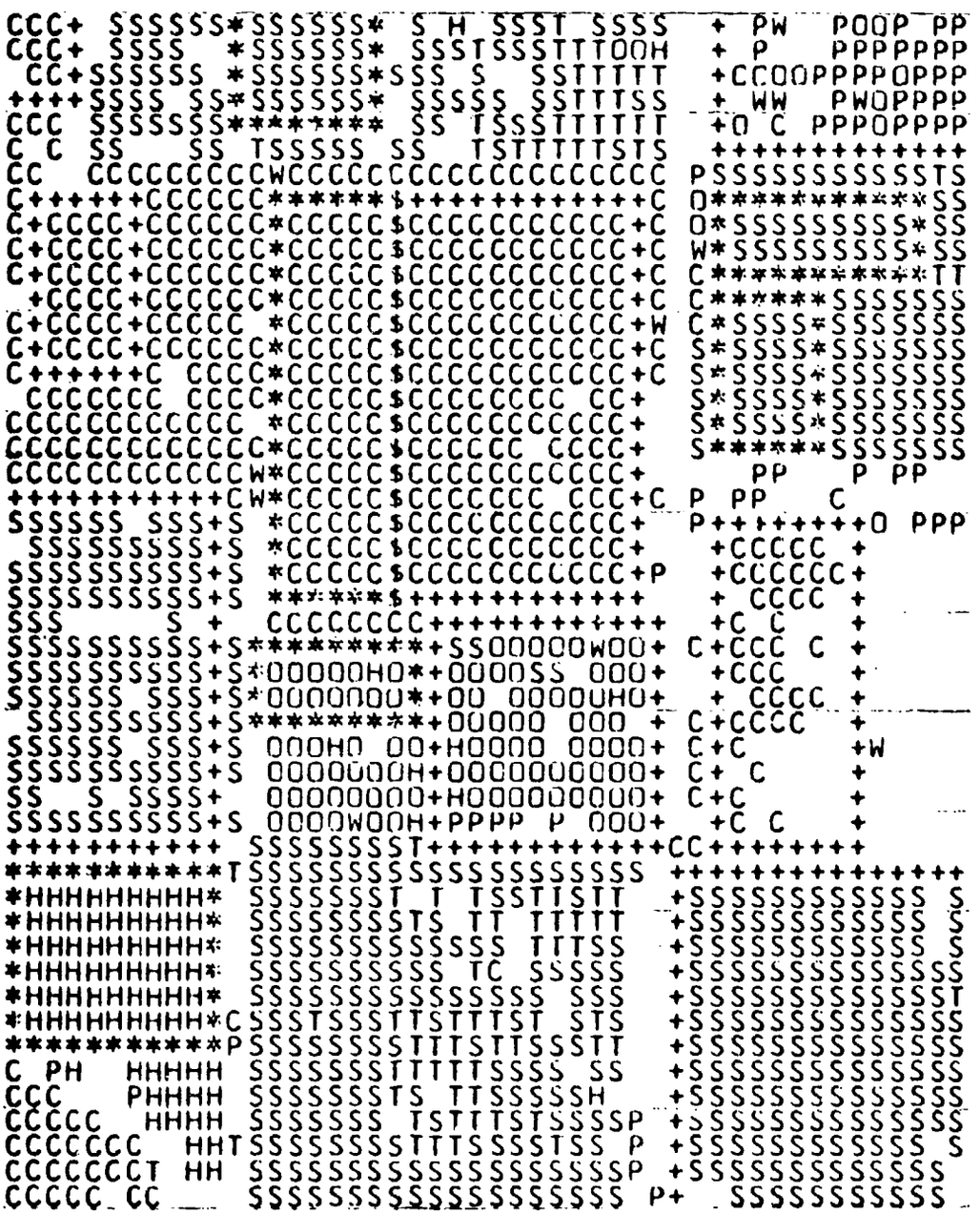


Figure 4 The classified result of the area of Fig. 1

STATISTICAL DEPENDENCY MODELS OF CONTEXT

P. H. Swain and T. S. Yu

In the previous quarterly report, we described a model for incorporating context in the image analysis process through compound decision theory. Briefly, under a fairly stringent set of assumptions a Bayesian strategy is employed which classifies a point into one of a candidate set of classes based on the multispectral data from the point itself and the data from either the neighboring four or neighboring eight points. Figure 1 shows the results of applying this approach for classifying a small set of LANDSAT multispectral scanner data. A block of imagery 128x128 pixels was classified using the "simple" rule (no context), with the 4-neighbor rule, and with the 8-neighbor rule. Samples of 900 pixels (30x30) were selected exhibiting a range of pointwise accuries and the corresponding accuracies obtained using the context-incorporating rules were tabulated together with the corresponding classification times. The results of the experiment show, as expected, the classifier using context is consistently better than the classifier which makes each decision based on data from the individual points. Furthermore, the "8-neighbor classifier" is consistently better than the "4-neighbor classifier".

However, the price paid in terms of computation time is substantial. To justify general use of this approach we shall have to demonstrate (a) its performance potential over a sufficiently broad range of analysis problems, and (b) a means of implementation (special purpose hardware, software, or a combination thereof) which is efficient enough to provide results on a cost-effective basis.

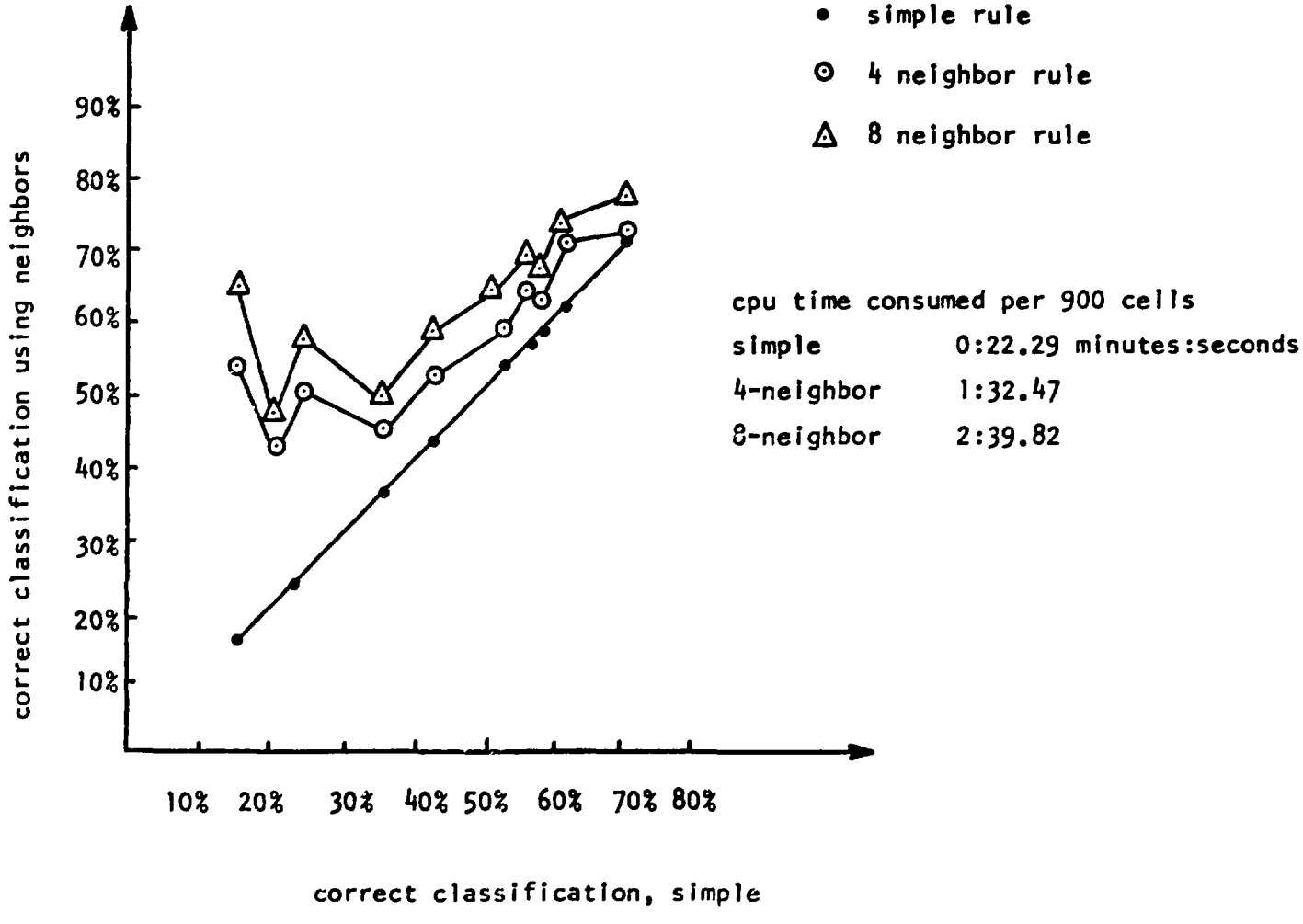


Fig. 1 Results of an experiment in utilizing statistical dependencies

IMAGE NOISE REDUCTION

M. Y. Yoo and T. S. Huang

I. INTRODUCTION

There have been many attempts to enhance images degraded by detector noise and imperfection of the imaging system [1-4]. In most cases we encounter two contradicting requirements: reducing the noise as much as possible, and retaining edge sharpness. The only reasonable answer is a compromise between the two. We will approach this problem using the synthetic highs technique where we can treat the smoothly varying part and the sharply changing boundaries separately and we may enjoy some freedom in compromising the two situations [5]. The performances of the system will be compared with several available heuristic approaches [6], [7].

II. IMAGE ENHANCEMENT SYSTEMS

2.1 Synthetic Highs Technique

This technique was proposed by Schreiber [8] and was used in two dimensional contour coding for data compression by Graham [9]. The basic idea of this technique is to decompose the image into two major elements (slowly varying "lows" and synthetic highs" which are mostly boundaries and textured parts) such that the recombination of the two elements results in the original again.

The technique was described in detail in previous reports and in Graham [9], and we are not going to repeat that here again. To use the technique for noise reduction, we need a noise reduction filter between the edge detector and the reconstruction filter. So we need an acceptable boundary detector for noisy images. Tang's [10] or the following simple edge detector may be used:

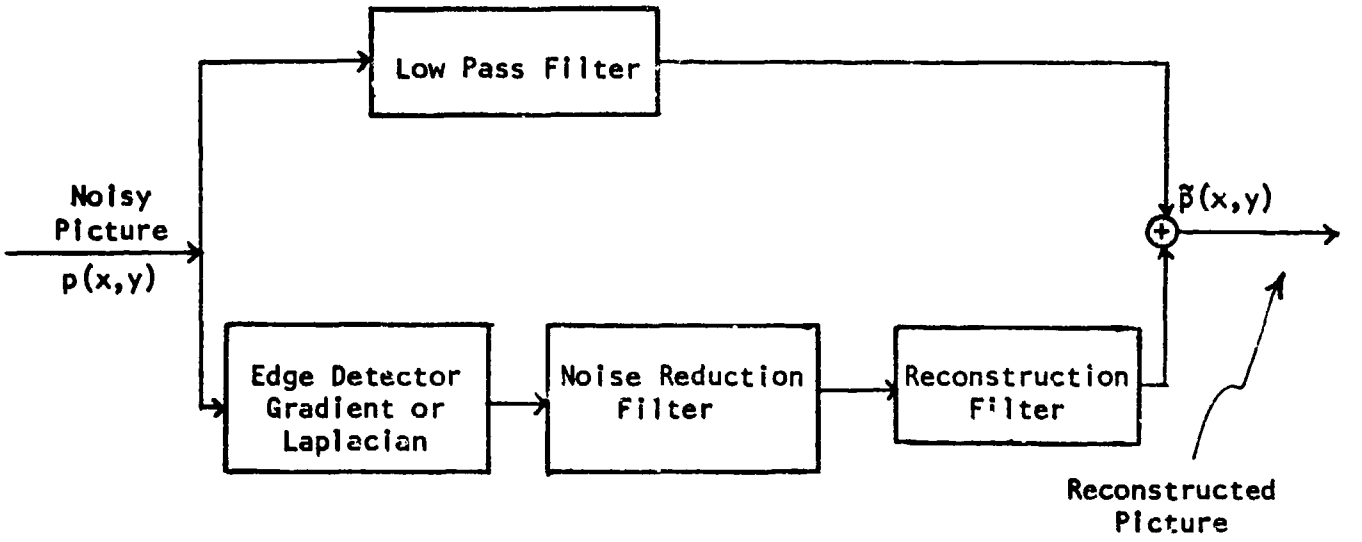


Figure 1 Synthetic highs technique

First we calculate the grey level differences in four different directions; horizontal, vertical, diagonal 1 (45^{o}), and diagonal 2 (135^{o}) as follows:

$$\text{Horz} = \frac{|G(x,y-\Delta y) - G(x,y+\Delta y)|}{G(x,y-\Delta y) + G(x,y+\Delta y)} \quad , \quad \text{Vert} = \frac{|G(x-\Delta x,y) - G(x+\Delta x,y)|}{G(x-\Delta x,y) + G(x+\Delta x,y)}$$

$$\text{diag 1} = \frac{|G(x-\Delta x,y+\Delta y) - G(x+\Delta x,y-\Delta y)|}{G(x-\Delta x,y+\Delta y) + G(x+\Delta x,y-\Delta y)}$$

$$\text{diag 2} = \frac{|G(x-\Delta x,y-\Delta y) - G(x+\Delta x,y+\Delta y)|}{G(x-\Delta x,y-\Delta y) + G(x+\Delta x,y+\Delta y)}$$

Whenever one of the following conditions holds we decide G(x,y) is on the boundary.

i) Horz $\geq \theta_1$ and Vert $< \theta_2$

ii) Horz $< \theta_2$ and Vert $\geq \theta_1$

iii) diag 1 $\geq \theta_1$ and diag 2 $< \theta_2$

iv) diag 2 $< \theta_2$ and diag 2 $\geq \theta_1$

where θ_1, θ_2 ($0 \leq \theta_1, \theta_2 \leq 1$) are preassigned threshold values. Typical values of θ_1, θ_2 are 0.2, 0.1.

2.2 Median Filter

We use an n x n (n is odd) window to scan the noisy image and replace the grey level of the center by the median grey level within the window. Pratt [6] used one-dimensional window to remove impulse noise. Resolution of the filtered image is highly dependent upon the size of the window and appropriate size should be chosen to retain reasonable boundaries. 3x3 and 5x5 windows were used for our experiment.

2.3 Variable Width Filter

First we take the gradient of the noisy image and divide the absolute gradient level into four different intervals. We assign the smoothing filter of an appropriate size to each interval such that the lower the absolute gradient level is, the larger the duration of the filters impulse response is. One dimensional Gaussian and median filters can be used in the horizontal and the vertical directions sequentially.

The basic idea of this approach is that we retain more boundaries where the absolute gradient level is high by using narrow smoothing filters and heavily smooth out slowly varying part by wide smoothing filters. Uniform or non-uniform subdivision of gradient levels may be used depending upon the distribution of gradient levels. The sizes of the filters used are 0, 3, 5, 7.

2.4 Noise Cheating Technique

This technique is a combination of two averagings with different window sizes. We average the noisy image using an m x m window and average the noisy picture again by an n x n (n > m) window. In the original paper [7] the authors quantize grey levels of "severely" averaged images using a quantum step that is at least four times the standard deviation of the averaged picture. But if the standard deviation of the averaged picture is large enough, the quantizing process wipes out everything and this really happened

in our case. So we may have to skip the quantizing step depending upon the size of the standard deviation. After averaging the noisy image with two different sizes of windows we replace each grey level in the "lightly" averaged image by the closest grey level among the corresponding eight surrounding image points in the "severely" averaged image.

The idea is that we retain the resolution of the "lightly" averaged picture, while we enjoy the reduction of noise level of the "severely" averaged image. The combining scheme is a kind of discrete maximum likelihood approach. $m = 2$, $n = 3$ were used for the experiment.

III. EXPERIMENTAL RESULTS AND CONCLUSION

White Gaussian noise was added to generate a 10 dB (variance of signal/variance of noise) noisy picture of size 256x256. The bandwidth of the low pass Gaussian filter in synthetic highs system is 0.116. The original noiseless picture is shown in Fig. 2 and the 10 dB noisy picture and "lows" of the noisy in Fig. 3 and Fig. 4, respectively. Figure 5 shows the noise reduction filter used in synthetic highs system. But this filter should be extended by a 3x3 window so as to pass both the positive and the negative parts of boundaries detected by Laplacian or gradient edge detector, otherwise we lose resolution significantly. Reconstructed pictures are given in Fig. 6 and Fig. 7. Outputs of the two-dimensional median filter are shown in Fig. 8 and Fig. 9. In 5x5 median filtered picture we lost resolution quite a bit. Median filters seem to be very effective for removing impulse errors (all dark spots have gone away in both outputs), but still retain Gaussian noises at an unpleasant level. Variable width Gaussian filters are truncated at 2 x standard deviation.

A one-dimensional filter was used sequentially in horizontal and vertical directions and when we apply the filter in the vertical direction we use data

already averaged in the horizontal direction. This may cause more reduction of noise but resolution is lost also. Actually we didn't smooth out at the top interval where the absolute gradient is highest. (Filter size was specified by 0 in section 2.3.) Comparing with the variable width Gaussian filter, the performance of the variable width median filter is very poor. In the noise cheating technique we first averaged the noisy picture by 2x2 window and replaced the grey levels at 4 picture points in the window by the averaged level. We did the same thing with 3x3 window and eliminated isolated picture blocks (Note: 3x3 window will have the same grey level after averaging) by simply replacing the center grey level by the surrounding grey levels. Since the whole block (2x2,3x3) has the same grey level, the output of the noise cheating technique has lots of square blocks. The performance of this technique seems the worst. The two-dimensional median filter is most economical and easiest to apply but the synthetic highs system gives the best result although it is most costly.

Some of the techniques can be modified for better performance. For example, we may use the noise reduction filter used in the synthetic highs system for variable width Gaussian or median filter and we may use moving 3x3 overlapping window replace the center grey level by the averaged level rather than assigning the same grey level to the whole block. Finally, we emphasize that the performance of noise reduction techniques depends upon the type of noises involved.

REFERENCES

[1] T. S. Huang and et. al., "Image Processing," Proc. IEEE, Vol. 59, No. 11, November 1971, pp. 1586-1609.

[2] -----, IEEE Proc., Vol. 60, Special Issue on Digital Image Processing, July 1972.

[3] W. K. Pratt, "Generalized Wiener Filtering Computation Techniques," IEEE Trans. on Comp., Vol. C-21, No. 7, July 1972, pp. 636-641.

[4] A. Habibi, "Two-Dimensional Bayesian Estimation of Images," Proc. IEEE, Vol. 60, No. 7, July 1972, pp. 878-883.

[5] M. Y. Yoo and T. S. Huang, "Noise Reduction in Photographic Images," RADC-TR-75-307, pp. 50-54.

[6] W. K. Pratt, "Median Filtering," USCIPI Report, pp. 116-122.

[7] H. J. Zweig, E. B. Barrett, and P. C. Hu, "Noise Cheating Image Enhancement," JOSA, Vol. 65, No. 11, November 1975, pp. 1347-1353.

[8] W. F. Schreiber, "The Mathematical Foundation of the Synthetic Highs Systems," Research Lab of Electronics, M.I.T., QPR, Vol. 68, January 1963, pp. 140.

[9] D. N. Graham, "Image Transmission by Two-Dimensional Contour Coding," Proc. IEEE, Vol. 55, No. 3, March 1967.

[10] G. Y. Tang and T. S. Huang, "Edge Extraction Technique," RADC-TR-75-202, August 1975.

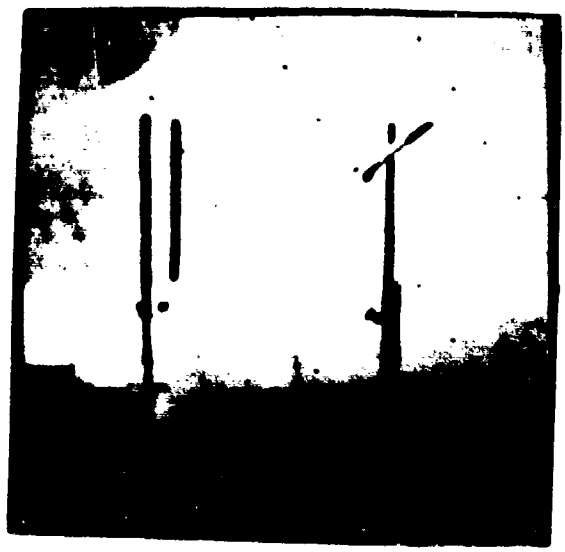


Figure 2 The original picture

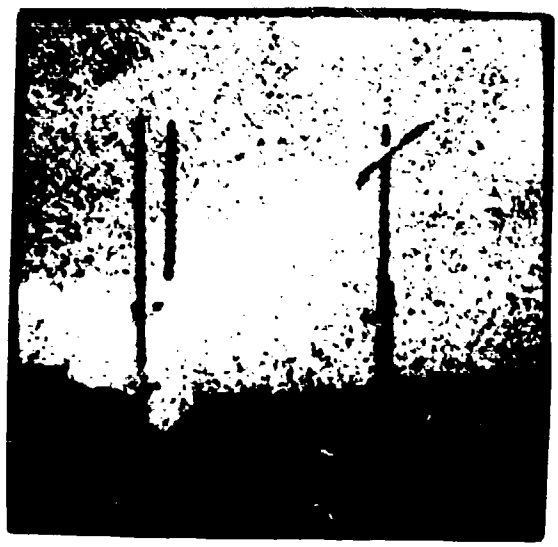


Figure 3 Noisy picture S/N = 10dB

Figure 4 Low pass picture
BW = 0.116

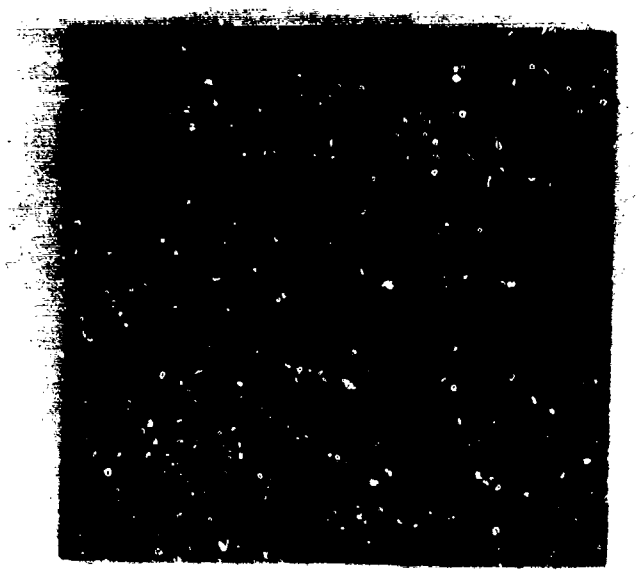


Figure 5 Noise reduction filter

Figure 6 Reconstructed picture
Laplacian synthetic
highs used

Figure 7 Reconstructed picture
Gradient synthetic
highs used

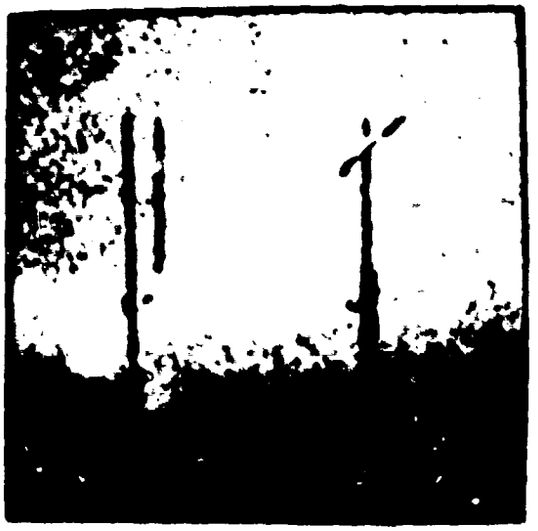


Figure 8 Reconstructed picture
3x3 median filter used

Figure 9 Reconstructed picture
5x5 median filter used

Figure 10 Reconstructed picture Variable width Gaussian uniform quantization used

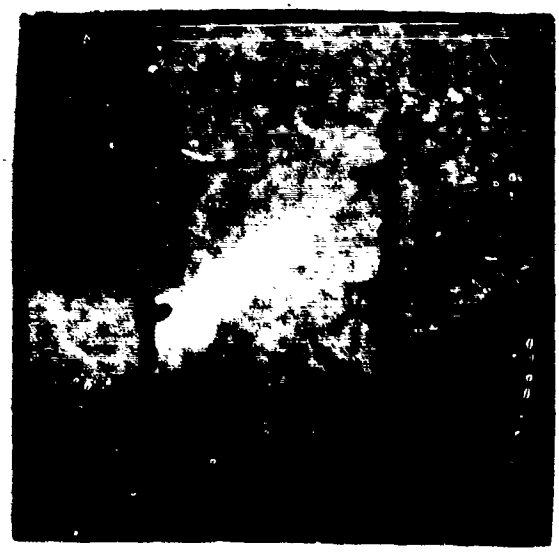


Figure 11 Reconstructed picture Variable width Gaussian non-uniform quantization used

Figure 12 Reconstructed picture Variable width median filter uniform quantization used

Figure 13 Reconstructed picture Variable width median filter non-uniform quantization used

Figure 14 Reconstructed picture
Noise cheating technique
2x2 and 3x3 windows used

TWO-DIMENSIONAL DIGITAL RECURSIVE FILTERS AND THEIR APPLICATION TO IMAGE PROCESSING

Brian O'Connor and T.S. Huang

Two dimensional digital recursive filters have the potential of saving computer time and storage in the processing of large two dimensional arrays such as images. In our research we are concerned with their design and their application to images. The desired processing to be performed on an image can generally be described in the frequency domain. In two dimensional filter design the filter coefficients are found which best approximate this desired frequency response, while at the same time guaranteeing stability.

Several possible design approaches exist. One relies on nonlinear optimization techniques to minimize the ℓ norm of the difference between the desired magnitude or group delay response and the filter response [1,2]. These techniques can be modified to guarantee stability. However, numerical problems arise with nonlinear optimization techniques, namely very large amounts of computation, sensitivity to starting points, and the possibility of converging to a local rather than global minimum. In addition, it is necessary to check stability of the designed filter at each iteration of the algorithm. However, a recently proposed method has eliminated need for the stability tests [3]. Another approach is based on an algorithm which allows the designer to specify an arbitrary magnitude-squared characteristic which is approximated optimally in a weighted Chebyshev (minimax) sense. Here, the optimization procedure can be formulated as a linear programming problem and the filter coefficients can be calculated using the two-dimensional discrete Hilbert transform to approximately factor the two-dimensional magnitude-square frequency response [4]. There are two problems in using this algorithm. The first is that fairly large amounts of computer time are needed to design filters. Secondly, the

factorization of magnitude-square function to obtain a filter implementation contains an infinite number of terms and hence truncation is necessary.

Generally, the nonlinear optimization techniques require that the filter consist of first and second order sections connected in cascade. This cascade form has many advantages over the direct form [1]. In our research we are investigating the approximation problem, i.e., how well can a cascade of first and second order sections approximate an arbitrary frequency response.

Another design approach is to find filter coefficients which approximate a desired frequency response without adding the stability constraint; then, if the filter is unstable, stabilize it so that the resulting filter has approximately the same frequency response. Several stabilization methods exist. The first was proposed by Shanks [5]. It stabilizes an unstable filter by finding its double planar least squares inverse (PLSI). Until recently the PLSI of a filter was assumed to be always stable. However, a special counter example has been constructed by Genin and Kamp where they find a 2x2 PLSI of a 4x4 array [6]. It is still an open question whether a PLSI of the same size as the input array is stable. Jury [7] has proved this for the special cases of 3x2 and 2x3. The double PLSI of an array will produce a filter array whose magnitude response approximates that of the unstable filter. However, in many applications the approximation is not adequate. In our work we have found an example where the double PLSI calculated by Read and Treitel is unstable. But no claims on finding a significant counter example will be made until Read and Treitel's [8] calculation of PLSI can be checked.

Another approach stabilizes an array by developing a two-dimensional discrete Hilbert transform to calculate the analytic phase function from the magnitude-squared frequency response. The method accomplishes stabilization with little accompanying distortion of its amplitude spectrum. Several developments

of the discrete Hilbert transform exist. Read and Treitel [8] generalized it to two dimensions by a direct extension of the one dimensional discrete case. The amplitude spectrum of the stabilized filter more closely approximates the desired magnitude than the double PLSI filter. However, not all filters produced by this method are stable. An example of this was given in their paper and we have found many more. Another method relies on discretizing the continuous two-dimensional Hilbert transform to obtain a two-dimensional discrete Hilbert transform. This has been derived by Dudgeon [9] and we have formulated and programmed a special case. Preliminary results seem to indicate that this method does produce stable filters, but the approximated amplitude is notable distorted.

We have been studying a third method which either employs the cepstrum or complex cepstrum. The study was motivated by work done by Ekstrom and Woods [10] on two dimensional spectral factorization. The inverse cepstrum of the first quandrant of the autocorrelated unstable filter's cepstrum is windowed to give a stable filter whose frequency response is close to that of the unstable filter. Preliminary results show that even though stability is not guaranteed the resulting filters are usually stable. This method can also be used to test stability of any two-dimensional recursive filter. The original filter is stable if the calculated array is equal to the original array. Because the FFT is used the correspondence is only approximate and some equality measurement must be used to ascertain stability. Through experimentation we found that for 3x3 arrays the filter is stable if the mean square difference between input and output arrays is less than .0000043. (16x16 FFT were used.)

Another possible way of checking stability and stabilizing unstable filters is to work with the complex [12] cepstrum of the filter array. By properly processing the cepstrum we can obtain arrays which are nonzero

in certain distinguished regions in the (Z_1, Z_2) plane, and thus, depending on the the type of filter, guaranteed to be stability. Many problems exist in one-dimensional cepstral analysis and they are increased in two dimensions. A two dimensional complex cepstrum program has been written using a modification of a new phase unwrapping algorithm which was developed by Tribolet [11]. Details about the theory and implementation of two-dimensional cepstrial analysis will be given in the next report.

Future work will include a detailed analysis of two-dimensional cepstral techniques applied to filter design and stabilization.

BIBLIOGRAPHY

[1] G. A. Maria and M. M. Fahmy, "An ℓ Design Technique for Two Dimensional Digital Recursive Filters," IEEE Trans. on ASSP, vol. ASSP-22, pp. 51-21, Feb. 1974.

[2] G. A. Maria and M. M. Fahmy, "ℓ Approximation of the Group Delay Response of One-and Two-Dimensional Filters," IEEE Trans. Circuits and Systems, vol. CAS-21, pp. 431-436, May 1974.

[3] P. Ramamoorthy and L. Bruton, "Frequency Domain Approximation of Stable Multi-Dimensional Discrete Recursive Filters," submitted to IEEE Trans. Circuits and Systems.

[4] D. Dudgeon, "Two-Dimensional Recursive Filter Design Using Differential Correction," IEEE Trans. on ASSP, vol. ASSP-23, pp. 264-267, June 1975.

[5] Shanks, Treitel, and Justice, "Stability and Synthesis of Two-Dimensional Recursive Filters," IEEE Trans. Audio and Electro., vol. AU-20, June 1972.

[6] Y. Genin and Y. Kamp, "Counter-Example in the Least-Square Inverse Stabilization of 2D Recursive Filters," Electronic Letters, vol. 11, pp. 330-331, July 24, 1975.

[7] B. D. O. Anderson and E. I. Jury, "Proof of Shank's conjecture for Low Degree Polynomials," submitted for publication in the IEEE Trans. on Acoustics, Speech and Signal Processing.

[8] R. Read and S. Treitel, "The Stabilization of Two-Dimensional Recursive Filters Via the Discrete Hilbert Transform," IEEE Trans. Geosc. Electron., vol. GE-11, pp. 153-160, July 1973.

[9] D. Dudgeon, "Two-Dimensional Recursive Filtering," Ph.D. Thesis, MIT, Cambridge, Mass. 1974.

[10] M. Ekstrom and J. Woods, "Two-Dimensional Spectral Factorization with Applications in Recursive Digital Filtering," IEEE Trans. ASSP, vol.-24, pp. 115-127, April 1976.

[11] J. Tribolet, "A New Phase Unwrapping Algorithm," submitted to IEEE Trans. on Acoustics, Speech and Signal Processing, March 1976.

[12] A. Oppenheim and R. Schafer, Digital Signal Processing, Prentice-Hall, Inc. 1975.

COMPARISON OF THE PROJECTION METHOD WITH SINGULAR VALUE DECOMPOSITION

S.P. Berger and T.S. Huang

The purpose of the work has been to evaluate the relative merits of the projection method of image restoration as compared with the singular value decomposition (SVD) approach.

The projection algorithm is an iterative method of solving a set of linear equations, where the equations are the discrete representation of the degradation process. The algorithm can incorporate a priori information. It has been shown to yield effective results for various types of degradation, including space-variant distortions.

The SVD approach has been used successfully in image restoration. It involves the treatment of the degradation in matrix form. By application of a pseudo-inverse matrix, the restoration is hopefully achieved.

Both approaches are limited by the effects of noise. The required number of iterations in the projection method and the number of terms utilized with SVD, are determined subjectively. The effects of noise increase with the number of terms and the number of iterations, and tend to overshadow the restoration process.

In the actual implementation of the SVD, difficulties with this approach have arisen. The amount of computer time required for the calculation of eigenvalues and eigenvectors can be prohibitive for large degradation matrices. Also, perhaps to the size of the matrix, the actual implementation has yielded faulty results. The success of the method depends heavily on the type of degradation that is effected on the border of the image. Minor changes in the form of the degradation matrix seem to create major problems in the operation of the computer implementation. The next report will contain at least a partial resolution of these difficulties.

FOURIER DESCRIPTORS

T. Wallace and P.A. Wintz

The Fourier descriptor (FD) is one method of describing the shape of a planar figure. Given a figure in the complex plane, the contour can be traced, yielding a complex function of time. If the contour is traced repeatedly, the periodic function which results can be expressed in a Fourier series. Granlund [1] defines the FD of a contour as the coefficients of this Fourier series.

To implement this method of shape description, it is necessary to sample the contour at a finite number of points. Since the discrete Fourier transform of a sequence gives us the values of the Fourier series coefficients of the sequence, assuming it to be periodic, using an FFT algorithm satisfies the definition above. The computational advantages of the FFT are well known.

The goal of this work is to classify the shapes of objects using their Fourier descriptors. The operations of rotation, scaling, and moving the starting point are easily implemented in the frequency domain by simple arithmetic on the frequency domain coefficients. While shapes may be compared in the space domain, the procedures required to adjust their size and orientation are computationally very expensive. Normally an iterative type of algorithm is employed, which searches for an optimum match between the unknown shape and the test set.

The goal of classification using Fourier descriptors is to develop an algorithm which will normalize the size and orientation of a shape before any comparisons to test shapes are made. If this can be accomplished, the classification process becomes a simple clustering problem with no iterative searches to contend with.

In our last quarterly report, the FD normalization problem was discussed, and the results of an experiment presented. It was shown that if the

contours under investigation were bilaterally symmetric, much less information was contained in the phases of the FD than in the magnitudes. The validity of classifying contours using the magnitudes of their FD's was demonstrated by classifying 18 airplane contours using this method. The Euclidean distance measure was used, and the distances between FDs proved to correlate well with the actual differences between the contours.

The goal of this study of Fourier descriptors is to eventually apply this method to contours traced using actual photographic data. To simulate this situation, an experiment was conducted using 20 aircraft contours digitized to two different resolutions, 128x128, and 64x64. The high resolution versions were taken to be accurate representations, and the lower resolution versions were assumed to be corrupted by noise and quantization error. Examination of representative contours (Figs. 1-6) show that the 128x128 contours are quite good representations, while the 64x64 show significant distortion of the smaller important features. This experiment was performed in order to test various distance measures, as well as to test suitability of the algorithm for use with actual photographic data.

While the experiment described in [1] was useful in demonstrating the general validity of classifying contours using distances between normalized Fourier descriptors (NFDS), a comparison of various distance measures was difficult due to lack of a definitive measure of similarity among the contours themselves. This obstacle was overcome by using the different resolution versions of the same planes.

The mean square criterion used previously was compared to an absolute value criterion. The results using the absolute value criterion were slightly better, as every 64x64 contour was correctly identified, whereas using the mean square criterion, only 19 out of 20 were correctly identified.

A simple application of Parseval's theorem shows that the mean square criterion in the frequency domain corresponds to a sum of the point by point mean square error in the time domain. There are some difficulties in mathematically relating the frequency domain absolute value criterion to the contours in the time domain. However, it is easy to compare the two criteria on a qualitative basis.

It is obvious that the absolute value measure will be more tolerant of one or two large coefficient differences than the mean square measure. Using either method, the largest coefficients will account for most of the distance measured. Accordingly, we should consider the possibility of large variations between coefficients of large expected value, which turn out to be the lowest frequency ones.

The second largest coefficient for each airplane FD is A(-3). The effects of varying this coefficient are to change the width of both the wings and the body of the plane. Smaller detail such as engine shape is virtually unaffected. Another coefficient of large expected value is A(-1). As discussed in [1], this coefficient describes the length to wingspan ratio of an airplane contour. More generally, varying A(-1) tends to elongate any contour. Again, the smaller detail is not greatly changed.

In summary, the absolute value criterion should tolerate more differences in gross structure of the contour, such as elongatedness or thickness to length ratio, while emphasizing variations in smaller detail. It seems likely that the absolute value measure might correspond more clearly to the differences which human observers find important than does the mean square criterion.

The classifications using the two methods were too similar to offer a definitive comparison of the two classification methods. However, the success of the experiment shows that we are ready to apply the FD algorithm to contours

extracted from actual photographic data. We plan to interface the BLOB contour tracing algorithm to the FD program as the next step in this research. The magnitude vs. phase information question will also be examined in the context of developing a normalization procedure which preserves all of the information contained in the contour.

REFERENCE

[1]. G. H. Granlund, "Fourier Preprocessing for Hand Print Character Recognition," IEEE Trans. on Computers, Vol. C-21, pp. 195-201, Feb. 1972.

Figure 1 128 x 128

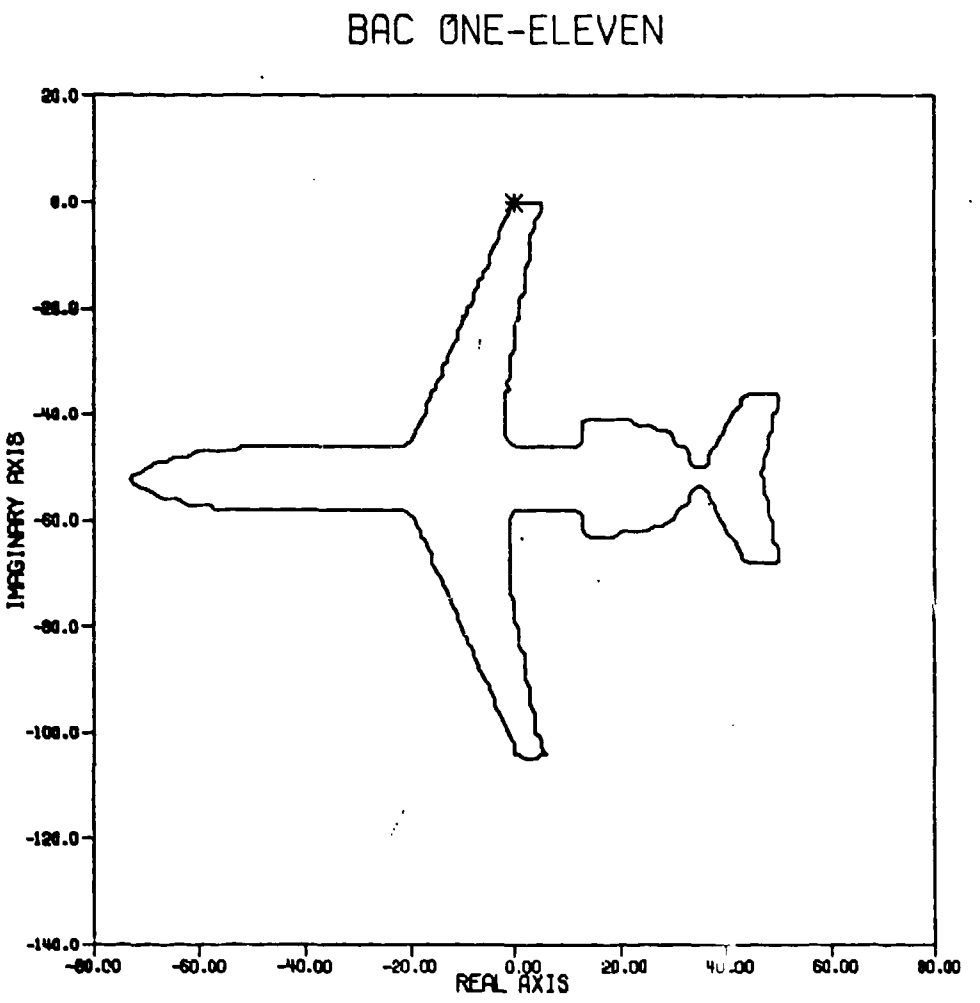


Figure 2 64 x 64

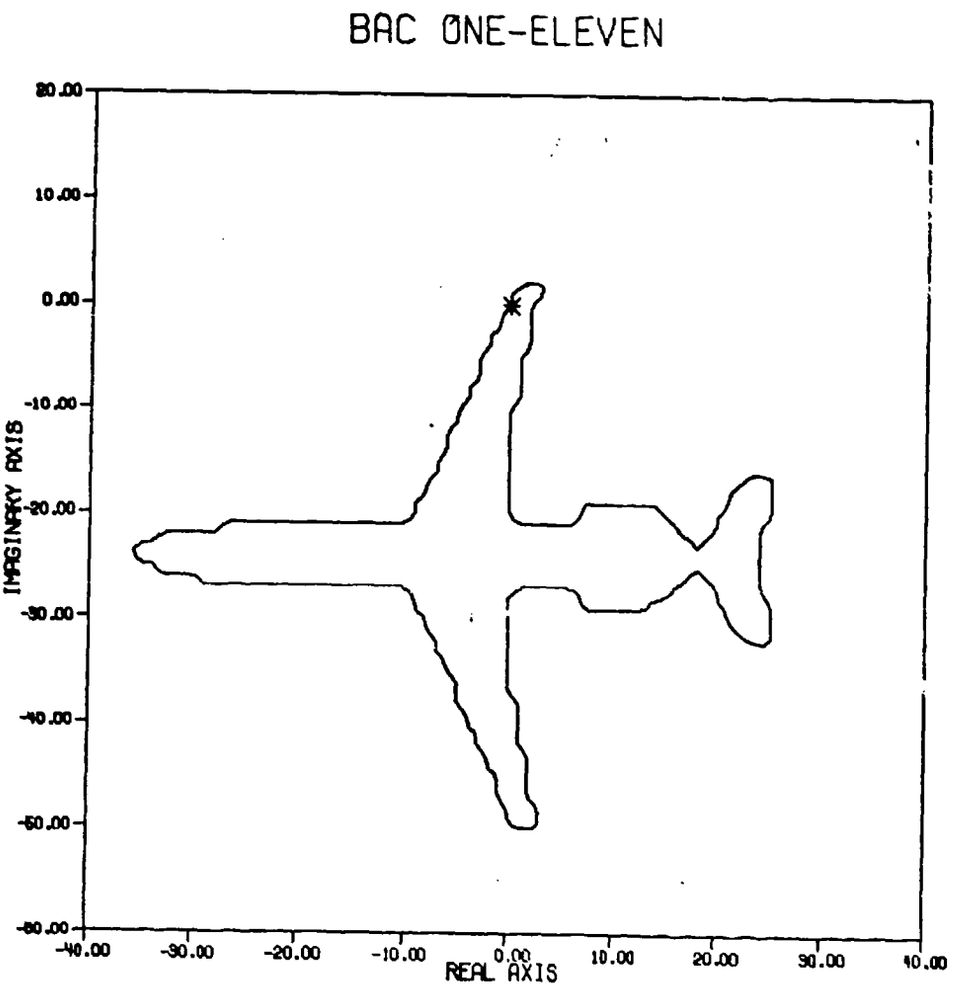


Figure 3 128 x 128

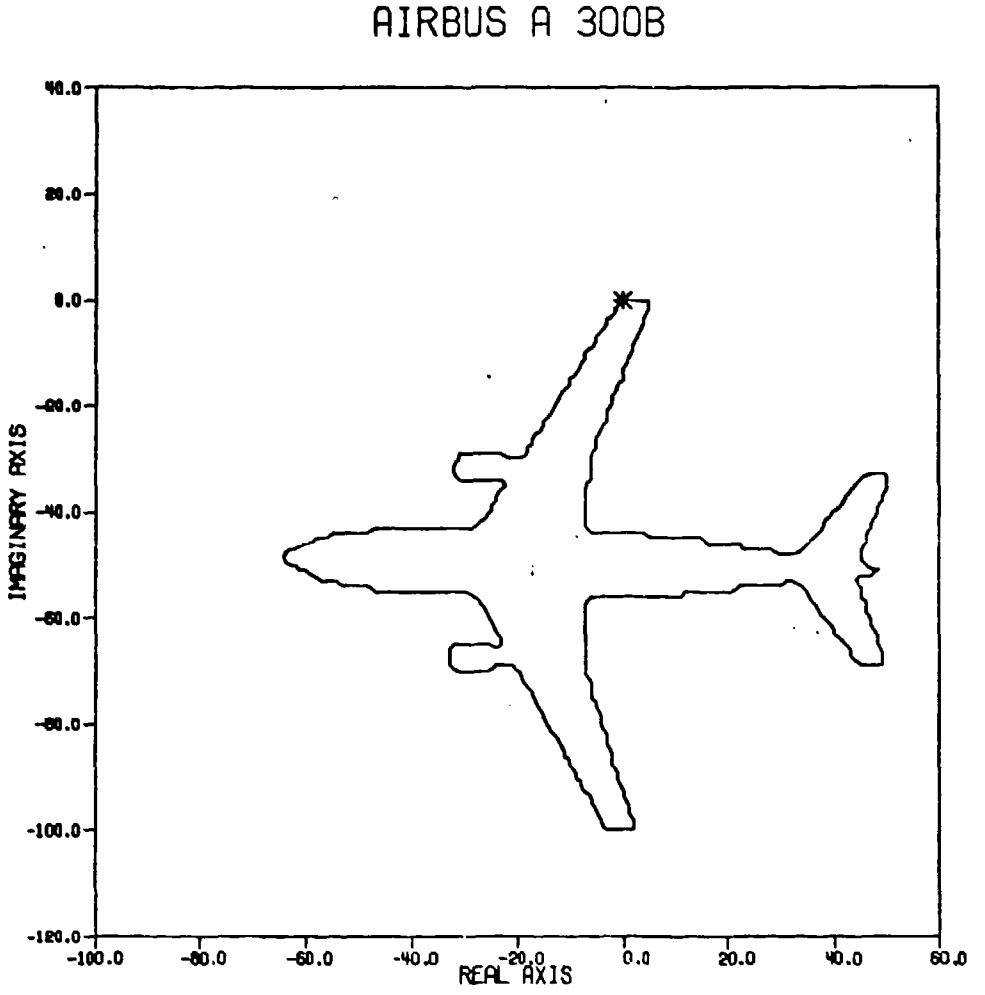


Figure 4 64 x 64

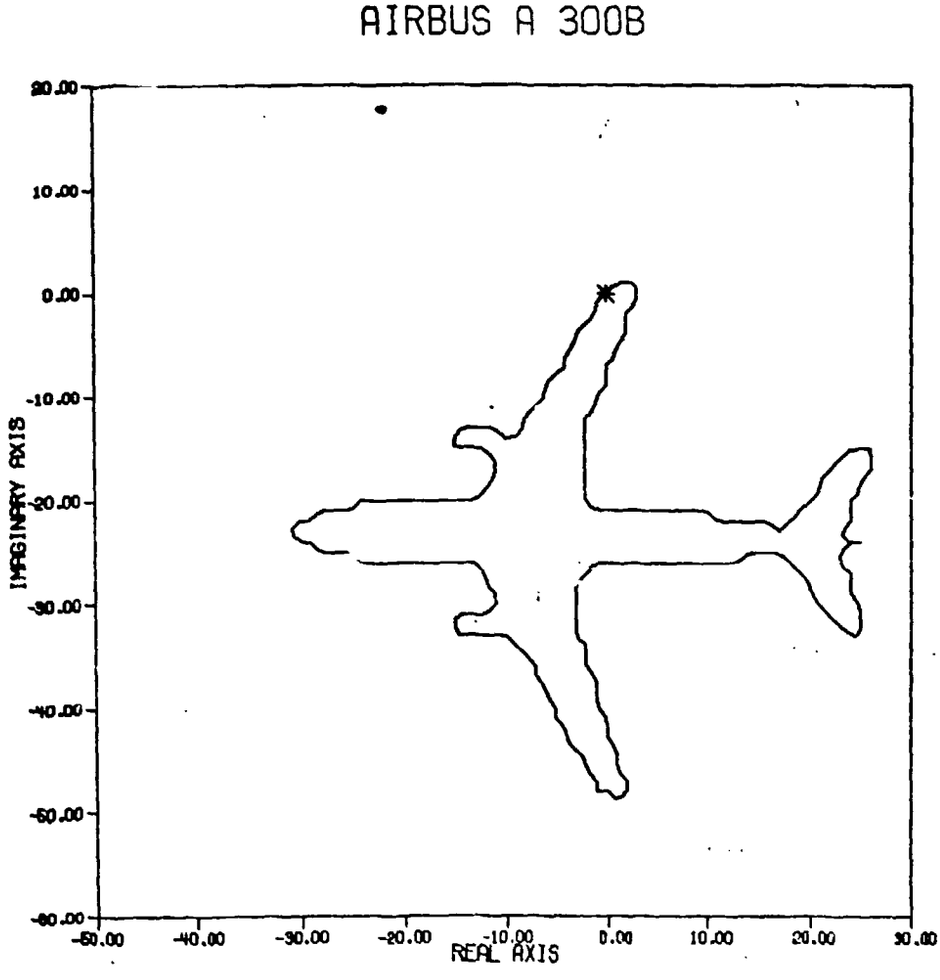


Figure 5 128 x 128

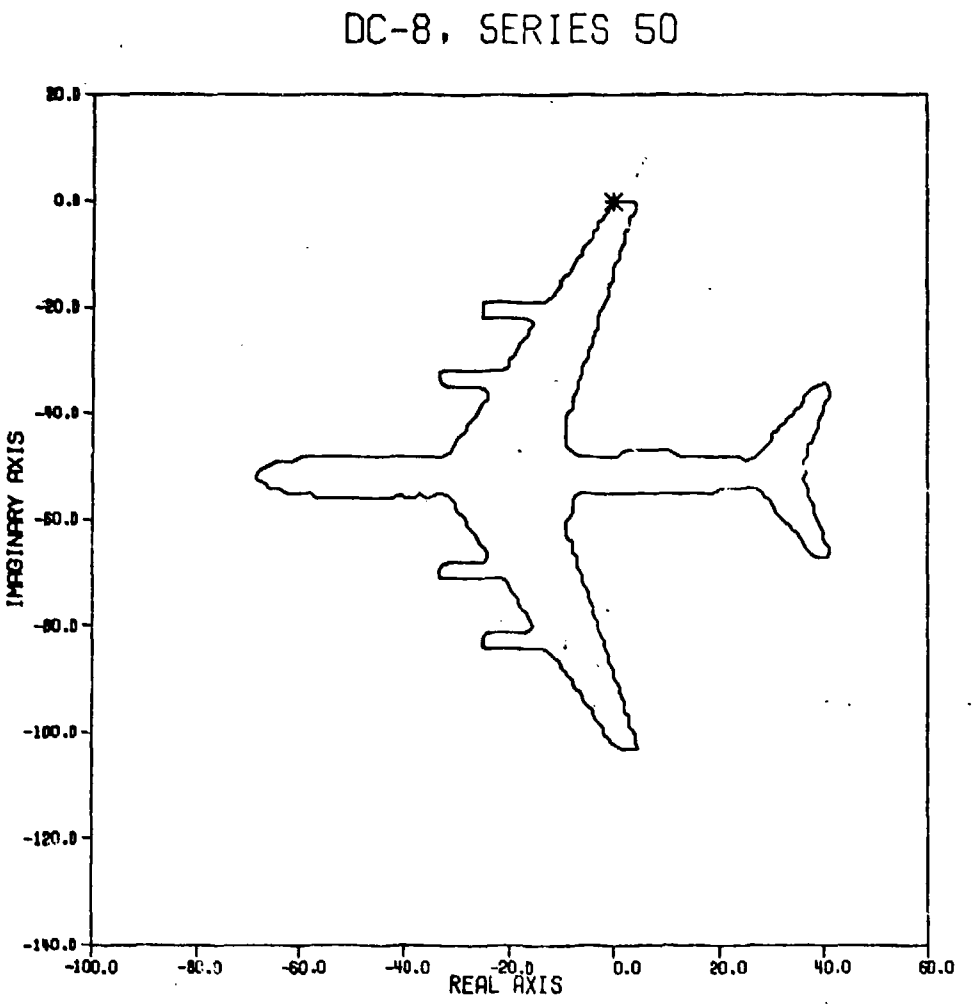
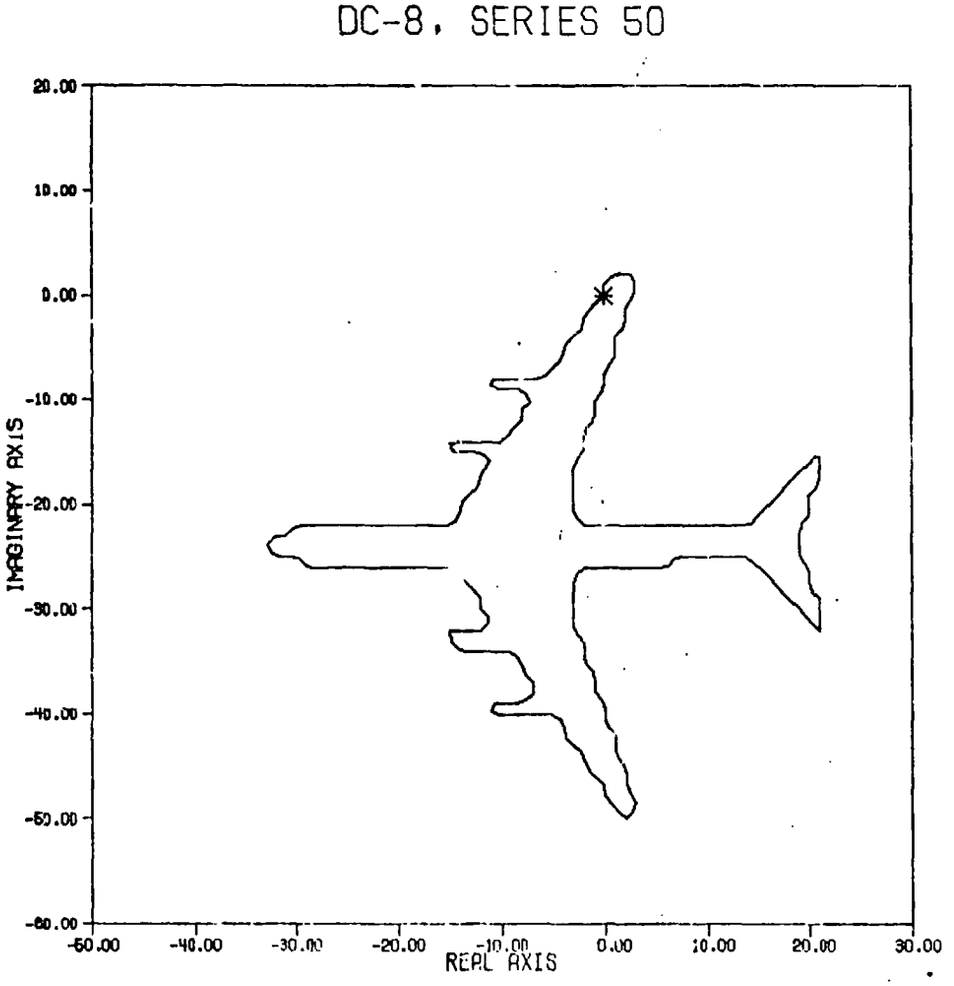


Figure 6 64 x 64



FILTERING TO REMOVE CLOUD COVER IN SATELLITE IMAGERY

O. R. Mitchell and P. L. Chen

I. INTRODUCTION

This recent work is a continuation of the previous report "Satellite Imagery Noise Removal" [1]. We are using 3-dimensional homomorphic filtering techniques to remove cloud cover in LANDSAT data.

In the previous filtering results the noise power spectrum was estimated by classifying each region of the noisy picture according to the level of noise present using the multispectral data analysis software system developed by Lars. These filtering results were not satisfactory due to the spectral indistinctability of clouds and concretes.

It may also prove impractical to use a multispectral classification program to find the noise statistics. Instead it may be possible to use a generalized cloud power spectrum derived by averaging many sample spectrums together.

II. INDIRECT ESTIMATION OF NOISE STATISTICS

The easiest way to estimate the general power spectrum of cloud is done by using data taken over water where the reflection is almost constant. In this case, the transformed scanner image (L is sun illumination, t is cloud transmission, r is ground reflection, and s is the received scanner image)

$$\text{Log } [L - s(x,y)] = \text{Log } L + \text{Log } t(x,y) + \text{Log } [L - ar(x,y)]$$

is reduced to

$$\text{Log } [L - s(x,y)] = \text{Log } [L - ar(x,y)] + K$$

where K is a constant, and the power spectrum obtained is that of noise except for the d.c. (0,0) frequency point.

This power spectrum should be circularly symmetric since clouds have no preferred orientation and should consist mainly of low spatial frequency components since clouds are relatively large and smooth functions compared to ground reflectance.

III. THREE DIMENSIONAL FILTERING

The real potential in the cloud filtering process is incorporating a third dimension, the spectral channels, forming a three dimensional reflection $r(x,y,z)$ and cloud transmission $t(x,y,z)$. The generalized linear filter thus employed is three dimensional $H(\mu,\nu,\rho)$ using three frequencies (two spatial and one spectral). Although there are only four points in the spectral dimension for LANDSAT data, the method has good promise, because most clouds follow a fixed response in the spectral dimension: cloud transmission increases with wavelength in a predictable fashion. When this information is incorporated into the filter (by means of the 3-D power spectrum) image variations which have the cloud spectral response are filtered out and image variations which do not follow the expected response of clouds in the spectral dimension are left in. The three dimensional filter, therefore, tends to reject all variations that are low frequency in the spatial dimension and follow the cloud spectral response in the third dimension.

Figures 1 to 4 are the 3-D power spectrum obtained from averaging the spectrum from three separate 64x64 regions of clouds over water. The ordinate is a log scale. Figure 1 represents the spectral d.c. slice (sum of all four spectral points), Fig. 2 represents the "one cycle per spectral band" slice, etc. One interesting observation we have made is that most of the cloud information is contained in slice 1 only and that information in slices 2, 3, and 4 (other than the d.c. point in each) represent ground reflectance information and should be left in the filtered output.

Figures 5 to 8 are the 3-D filter functions obtained for one particular 64x64 section of the image taken over land. The general cloud spectrum was normalized so that it fell below the spectrum of "signal plus noise" and was then subtracted from it for the signal spectrum estimate. Note the filter tends to attenuate low frequencies in slice one but leaves them in slices 2 through 4.

Figure 9 shows the composite filtered for 21 64x64 blocks of LANDSAT data scanned over the Chicago area on a somewhat cloudy day.

IV. CONCLUSIONS

Three dimensional filtering of multispectral data to remove light cloud cover is a distinct possibility.

In Fig. 9 the filtered picture shows more detail than the original cloudy one. Lake boundaries and highways have clearer appearances in the filtered pictures.

The model of the cloud distortion needs to be refined based on the results of filtering using the simple model presented in the previous report. It may be necessary to consider convolutional effects of cloud cover as well as multiplicative effects. The change in multispectral classification accuracy after filtering may be a suitable measure of the performance of such homomorphic filter.

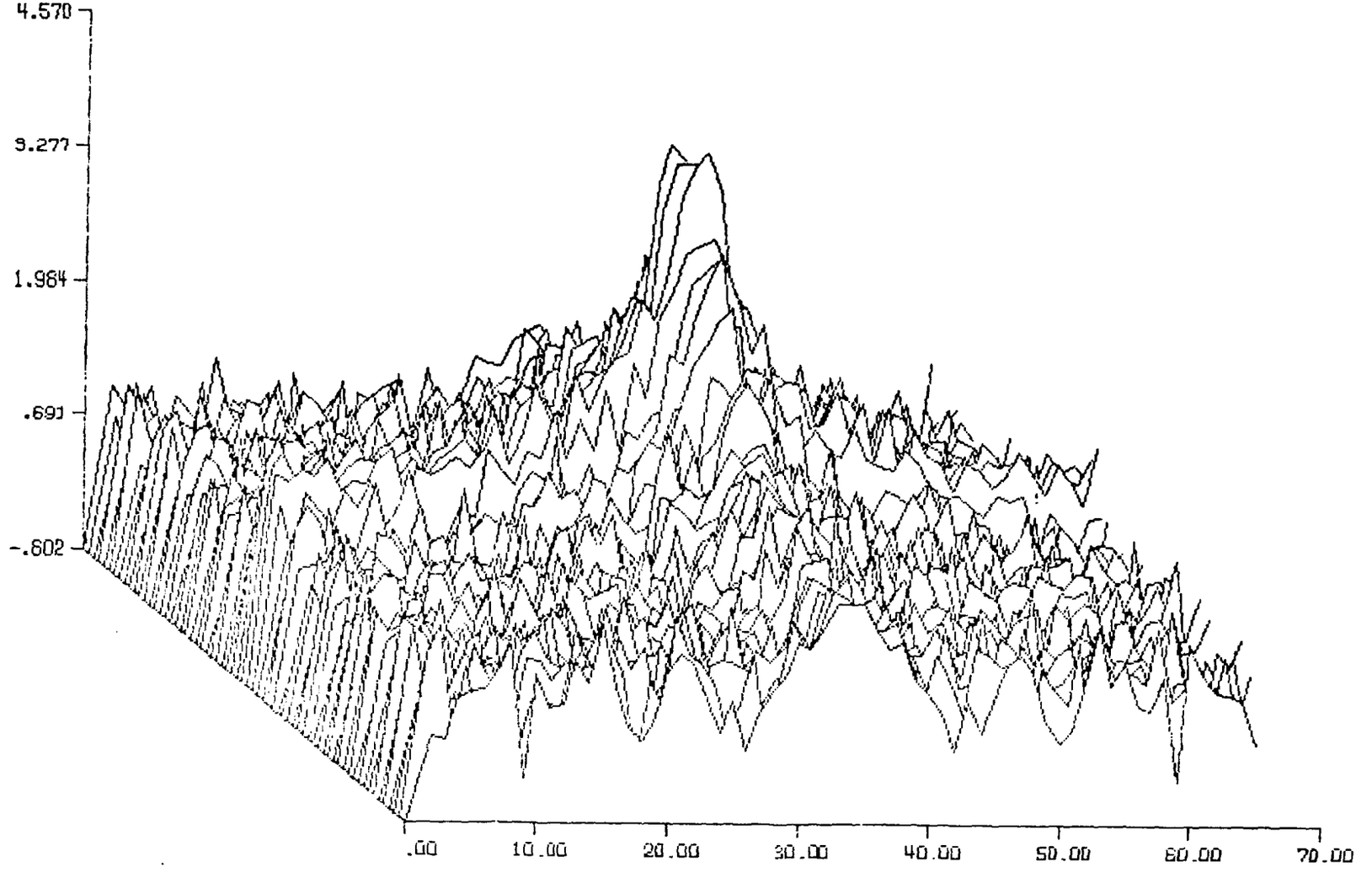


Figure 1 3-D cloud power spectrum - slice 1

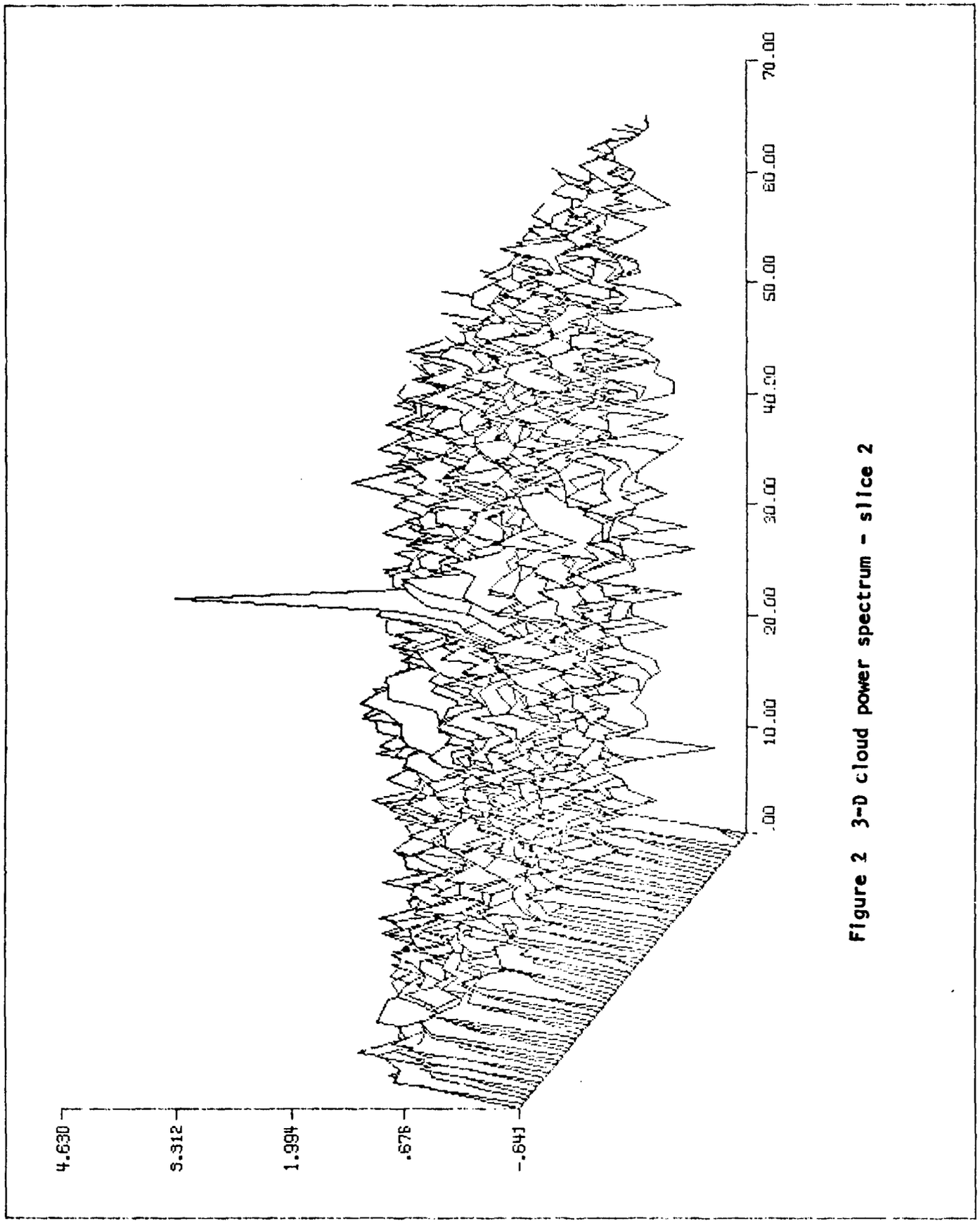


Figure 2 3-D cloud power spectrum - slice 2

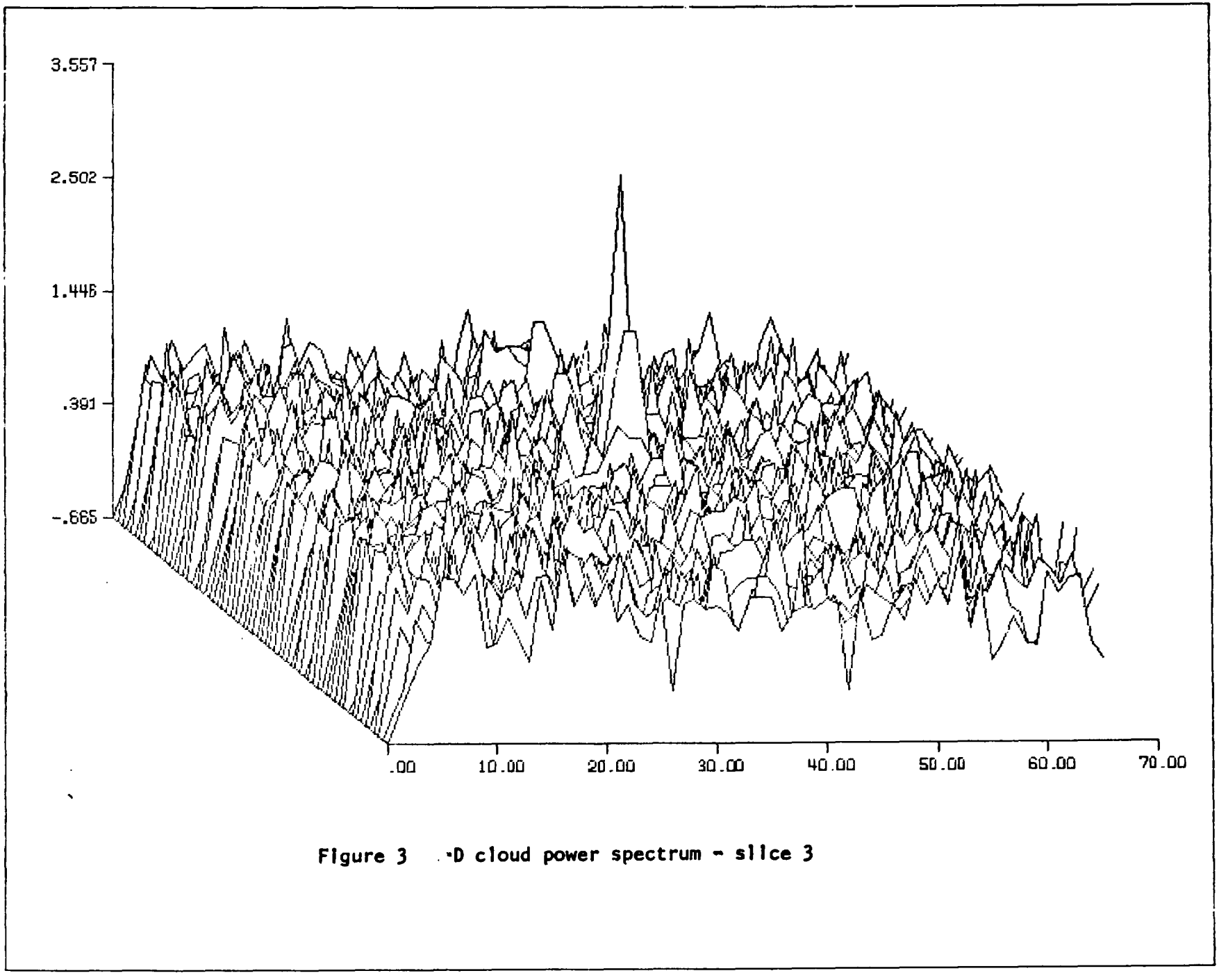


Figure 3 -D cloud power spectrum - slice 3

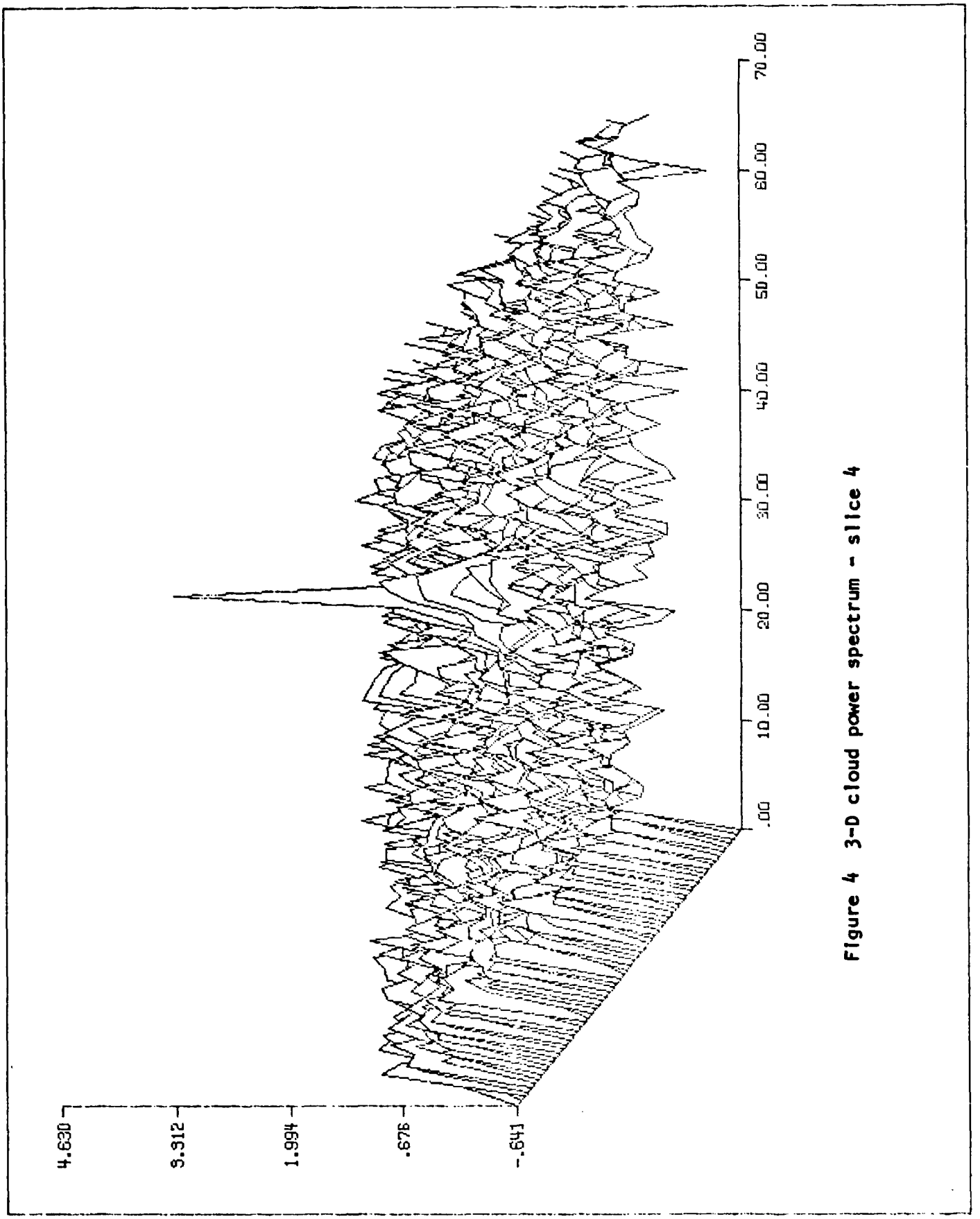


Figure 4 3-D cloud power spectrum - slice 4

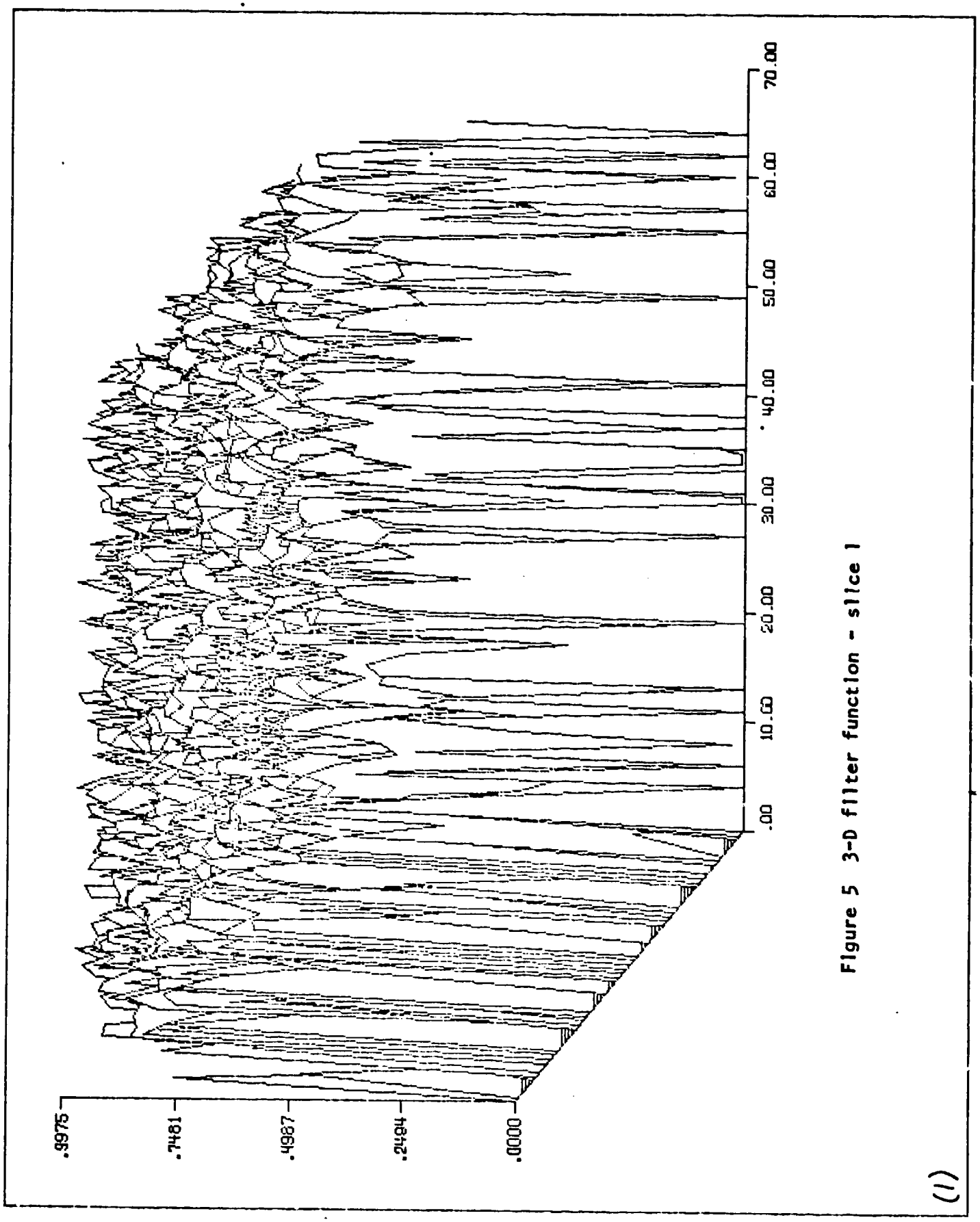


Figure 5 3-D filter function - slice 1

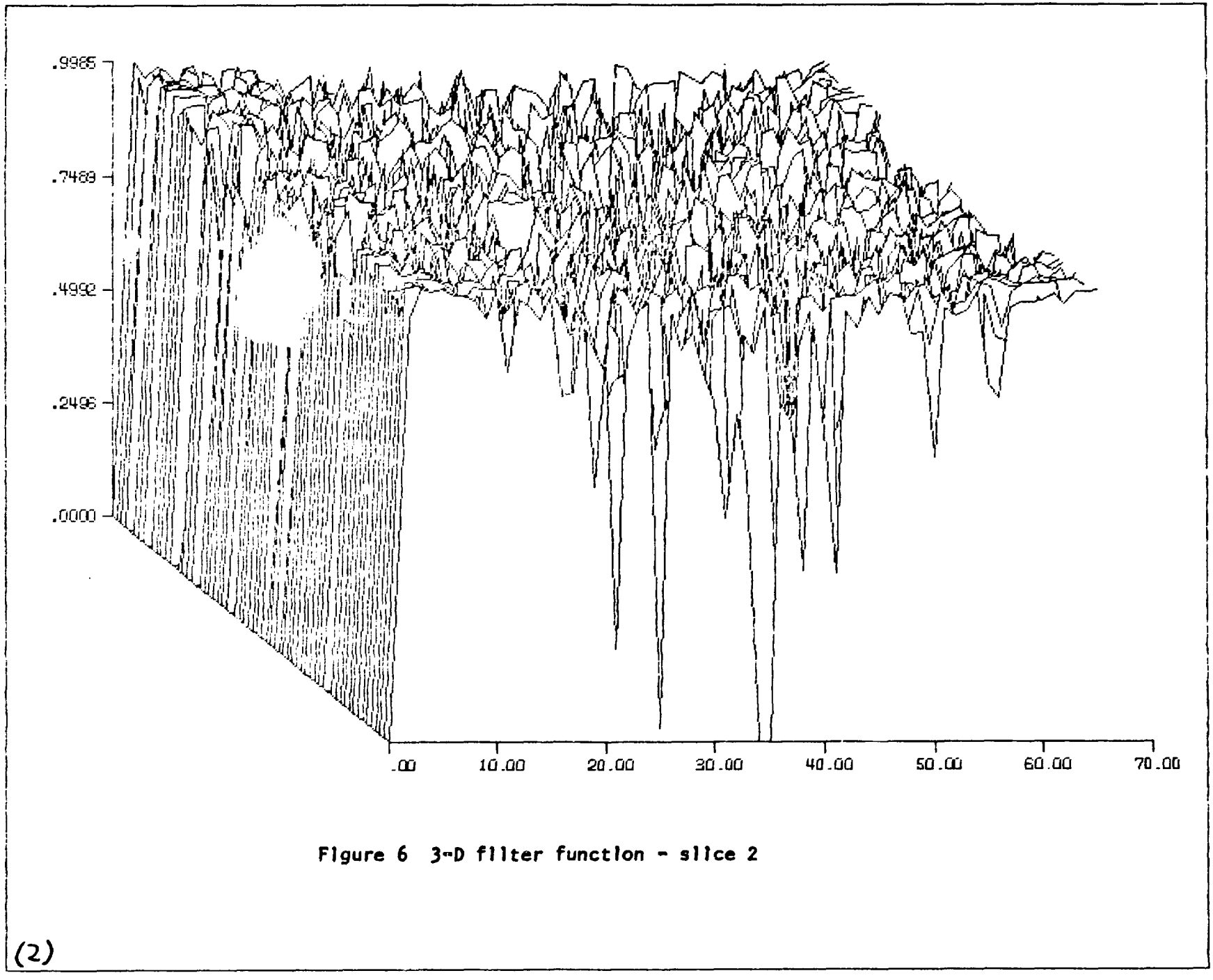


Figure 6 3-D filter function - slice 2

(2)

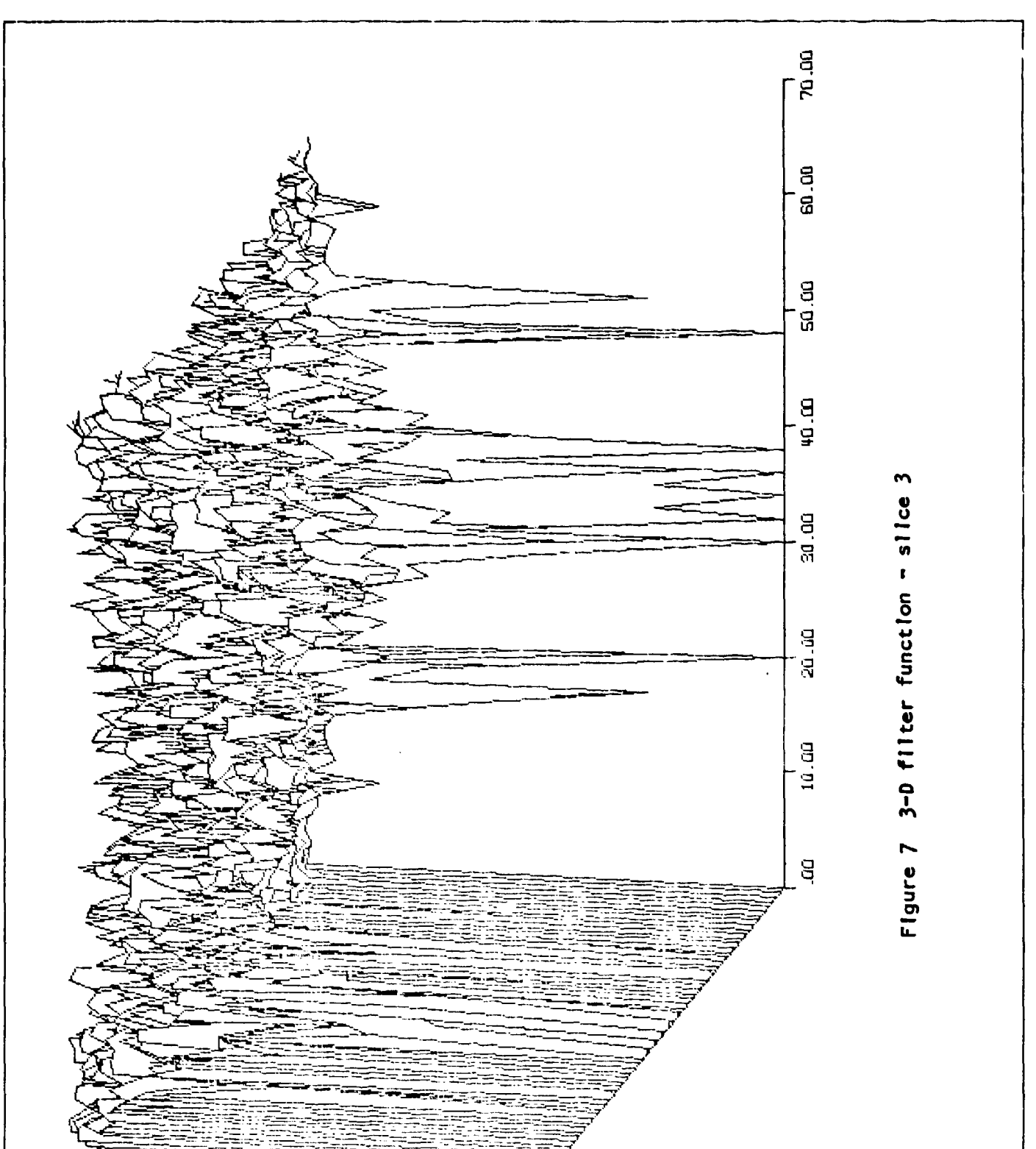


Figure 7 3-D filter function - slice 3

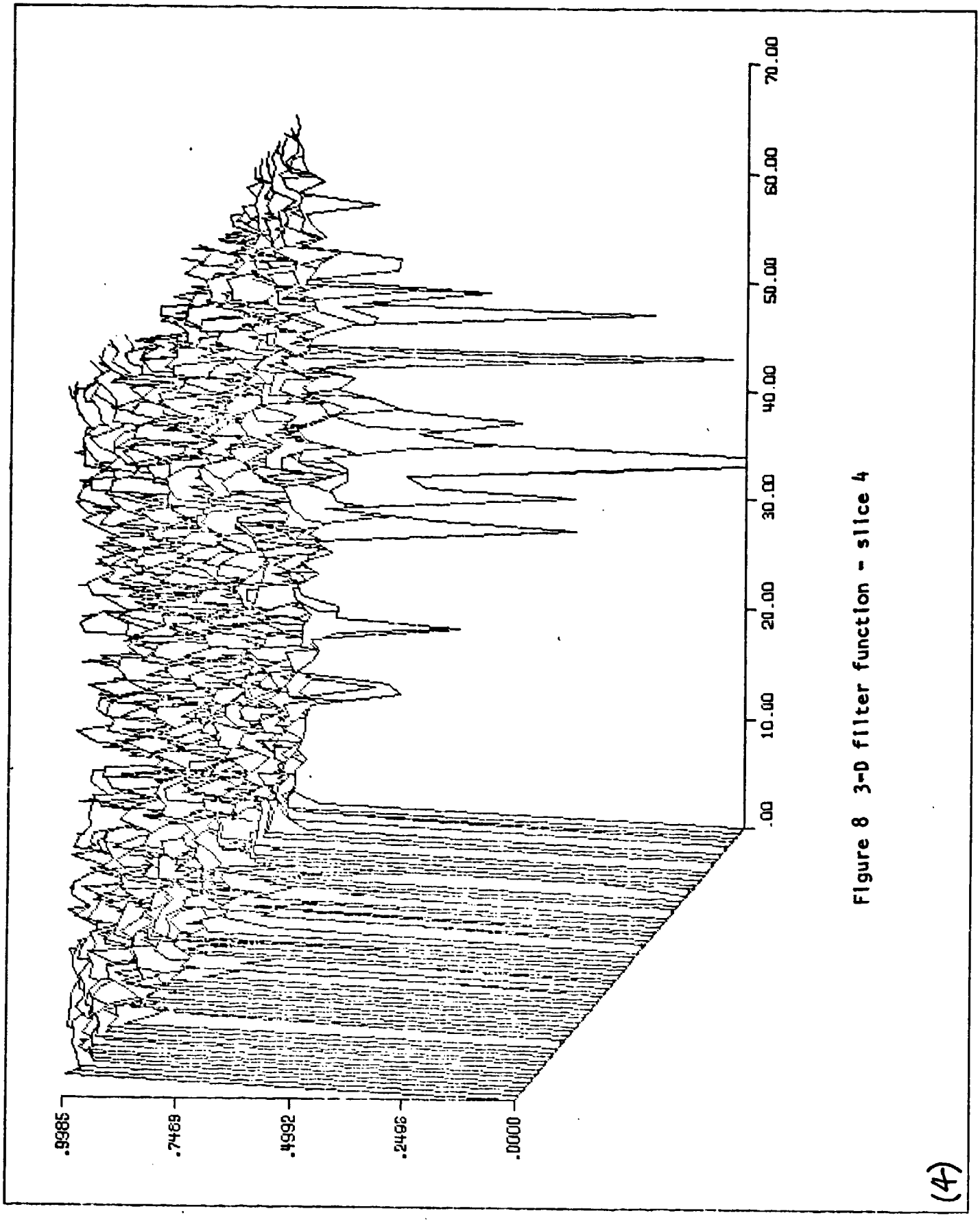


Figure 8 3-D filter function - slice 4

Figure 9(a) (Left) 3-D filtered output (channel 3)
(Right) Original Landsat data (channel 3)

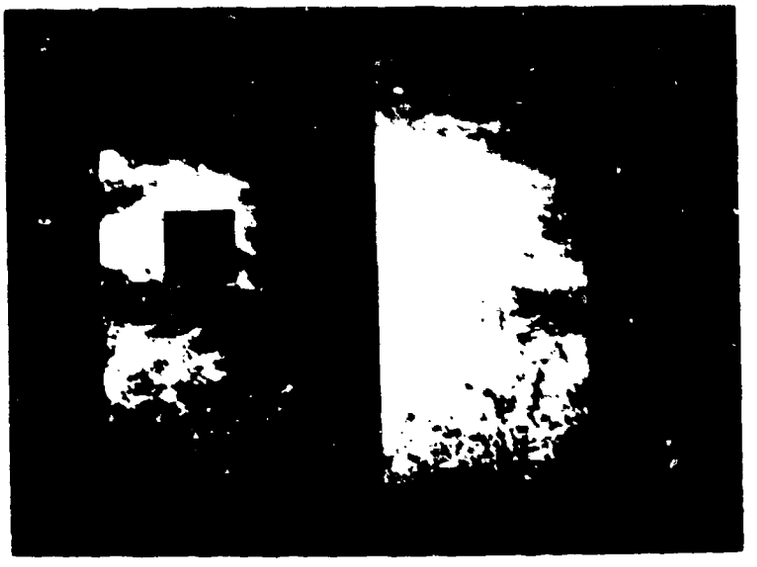


Figure 9(b) (Left) 3-D filtered output (channel 4)
(Right) Original Landsat data (channel 4)

FACILITIES

QTY	Manufacturer	Description
3	Beehive Elect.	"Super-Bee" Terminals
2	Tex. Inst.	"Silent 700" Terminals
1	Digi-Data	Industry standard magnetic tape system; 2, 9-track and 1, 7-track drives; one each NRZI and phase-encoded formatters/controllers
1	DEC	Dual-drive DECtape unit
1	DEC	RP03 disk drive (40 million characters)
1	Fabritek	96K-word auxiliary memory system (64K bought by ARPA, 32K by NASA)
1	Versatek	Electrostatic matrix printer
1	Comtal	Color picture display
1	Data Printer	132 column, 600 L.P.M. line printer
1	True-Data	Punched card reader
1	Tektronix	Model 4010, graphics display
1	DEC	PDP-11/45 computer system; system includes: 32K memory FPP-11 floating point processor (NSF money) H960 extension mounting cabinet 3 - small peripheral mountings blocks (DD-11) 1 UNIBUS repeater/expander DH11, 16-line terminal multiplexor KW11-p programmed clock "ANTS" - type PDP-11/IMP interface

Note: Our PDP-11/45 is currently operating under the UNIX system.

BOOKS

FU, K.S., "ERTS Data Analysis," chapter in Application of Syntactic Pattern Recognition, ed., Springer-Verlag, 1976 (with J. Brayer, P. H. Swain)

SWAIN, P.H., "ERTS Data Analysis," chapter in Applications of Syntactic Pattern Recognition, K. S. Fu, ed., accepted for publication by Springer-Verlag, 1976.

WINTZ, P.A., "Picture Coding and Feature Extraction," chapter in Digital Image Processing, 1976.

JOURNAL PUBLICATIONS

FU, K.S., "A Tree System Approach for Fingerprint Pattern Recognition," IEEE Transactions on Computers, Vol. C-25, pp. 262-274, March 1976 (with B. Moayer)

FU, K.S., "An Application of Stochastic Languages to Fingerprint Pattern Recognition," Pattern Recognition, Vol. 8, pp. 175-181, 1976 (with B. Moayer)

FU, K.S., "A Minicomputer Facility for Picture Processing and Pattern Recognition Research," COMPUTER, Vol. 9, pp. 70-77, May 1976 (with E. Persoon)

FU, K.S., "Parametric Feature Extraction Through Error Minimization, Applied to Medical Diagnosis," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-6, September 1976 (with T. Lissack)

MITCHELL, O.R., "Effect of Spatial Frequency on the Visibility of Unstructured Patterns," J. Opt. Soc. Am., Vol. 66, No. 4, April 1976.

MITCHELL, O.R., "Digital Communications Equipment for Instructional Purposes," IEEE Trans. on Education, May 1976 (with W. L. Thomas)

SWAIN, P.H., "Determining Density of Maize Canopy from Digitized Photographic Data," Agronomy Journal, Vol. 68, pp. 55-59, January-February 1976 (with E. R. Stoner and M. F. Baumgardner)

SWAIN, P.H., "Selective Radiant Temperature Mapping Using a Layered Classifier," IEEE Trans. Geoscience Electronics, Vol. GE-14, pp. 101-106, April 1976 (with L. A. Bartolucci and C. L. Wu)

CONFERENCES

FU, K.S., "Processing of Chest X-Ray Images by Computer," IFIP Working Conference on Decision-Making and Medical Care, May 24-29, 1976, Dijon, France.

FU, K.S., "A Syntactic Approach to the Representation of Image Structures," IFIP Working Conference on Environmental Modelling, April 26-28, 1976, Tokyo, Japan.

FU, K.S., "High Dimensional Languages and Grammatical Inference," IEEE Joint Workshop on Pattern Recognition and Artificial Intelligence, June 1-3, 1976, Hyannis, MA.

FU, K.S., "Some Applications of Stochastic Languages," Symposium on Application of Statistics, June 14-18, 1976, Dayton, Ohio.

FU, K.S., "Tree System Approach for LANDSAT Data Interpretation," Proc. Symp. on Machine Processing of Remotely Sensed Data, June 29-July 1, 1976.

FU, K.S.,"An Approach to the Design of a Linear Binary Tree Classifier," Proc. Symp. on Machine Processing of Remotely Sensed Data, June 29-July 1, 1976.

FU, K.S., "The Linguistic Approach to Pattern Recognition," Advanced Seminar on Classification and Clustering," The Mathematics Research Center, University of Wisconsin, Madison, Wisconsin, May 3-5, 1976.

HUANG, T.S., "Restoration of Images Degraded by Spatially-Varying Systems," presented at the OSA Technical Meeting on Image Processing, Asilomar, February 24-25, 1976.

HUANG, T.S.,"Nonlinear Estimation of Markov Jump Processes," presented at the IEEE Int'l Information Theory Symposium, Ronneby, Sweeden, June 21-24, 1976 (with J. Burnett)

HUANG, T.S., "Digital Straight Edges," presented at the 6th Annual Symp. on Automatic Imagery Pattern Recognition, Univ. of Maryland, Silver Spring, MD, June 1-2, 1976 (with G. Tang)

HUANG, T.S., "Two-Dimensional Fourier Transforms," "Image Restoration," and "Film Models," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

HUANG, T.S., "Some Experience in Stability Tests for Two-Dimensional Recursive Digital Filters," Arden House Workshop on Digital Signal Processing, Feb. 23-26, 1976, Hamison, NY (with B. T. O'Connor)

HUANG, T.S. and MITCHELL, O.R., "Subjective Effect of Two-Dimensional Noise," SPSE, Symp. on image Evaluation, July 19-23, 1976, Toronto, Canada.

HUANG, T.S., "Image Processing Research at Purdue," presented at Los Alamos Scientific Laboratory, Los Alamos, NM, May 19, 1976.

HUANG, T.S. and FU, K.S., "Research on Image Understanding at Purdue," ARPA workshop on Image Understanding, April 12-13, 1976, Univ. of Southern California, Los Angeles.

HUANG, T.S., organizer and lecturer, short courses on "Digital Techniques in Spectral Analysis, Estimation and Filtering," March 8-12, 1976, and "Image Processing and Pattern Recognition," April 12-16, 1976, Purdue University.

MITCHELL, O.R., "Texture Edge Detection and Classification Using Max-Min Descriptors," Sixth Annual Symposium on Automatic Imagery Pattern Recognition, Univ. of Maryland, College Park, MD, June 1-2, 1976.

MITCHELL, O.R., "Filtering to Remove Cloud Cover in Satellite Imagery," LARS Symposium, Machine Processing of Remotely Sensed Data, June 29-July 1, 1976, West Lafayette, IN (with P. L. Chen)

MITHCELL, O.R. and HUANG, T.S., "Subjective Effect of Two-Dimensional Noise," SPSE Symp. on Image Evaluation, July 19-23, 1976, Toronto, Canada.

SWAIN, P.H., "Application of a Class of Sequential Classifiers to Multitemporal Remote Sensing Data," Proc. Third Symposium on Machine Processing of Remotely Sensed Data, June 1976 (with H. Hauska)

SWAIN, P.H., "Some Time for Texture in the Spectrum of Spatial Features," presented at the Engineering Foundation Conference on Algorithms for Image Processing, Franklin Pierce College, Rindge, NH, August 1976 (with D. A. Landgrebe)

WINTZ, P.A., "Images and Models for Image Noise," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

WINTZ, P.A., "Image Coding with Emphasis on Techniques for Producing Decorrelated Image Data," presented at NATO Advanced Institute on Digital Image Processing and Analysis, Bonas, France, June 14-25, 1976.

STAFF

CO-PRINCIPAL INVESTIGATORS

T. S. Huang
K. S. Fu

PROFESSORIAL

K. Fukunaga
O. Mitchell
P. Swain
P. Wintz

GRADUATE RESEARCHERS

S. Berger
J. Burnett
Wm. Chan
P.H. Chen
P.L. Chen
X. Dang
R. Florek
J. Keng
R.L. Li
P. Narendra
B. O'Connor
D. Panda
A. Salahi
G. Tang
T. Wallace
M. Yoo
T.S. Yu

RESEARCH STAFF

J. Besemer
W. Robey

UNDERGRADUATE RESEARCHERS

C. Buckwacter
M. DeMoney
R. Johnson
J. Schwab

ELECTRONIC TECHNICIANS

D. Azpell

SECRETARIES

M. Barbour
M. Claire